

一套配教学视频的项目开发案例类丛书，累计销售160000册



10套
完整项目
开发案例

网站、QQ等技术支持，让学习畅通无阻

20小时
教学视频





模块库、题库、素材库登录本书网站



辛洪郁 张鑫 编著

PHP 项目开发全程实录 第3版

- 九九度供求信息网（视频讲解：1小时50分）
- BCTY365网上社区（视频讲解：2小时25分）
- 办公自动化管理系统（视频讲解：1小时52分）
- 网络在线考试系统（视频讲解：1小时22分）
- 物流配送信息网（视频讲解：1小时26分）
- 学校图书馆管理系统（视频讲解：2小时20分）
- 博客管理系统（视频讲解：1小时16分）
- online影视365网（视频讲解：1小时32分）
- 明日知道网站（视频讲解：1小时20分）
- 电子商务网站（视频讲解：2小时5分）

清華大學出版社

软件项目开发全程实录

PHP项目开发全程实录
（第3版）

辛洪郁　张　鑫　编著

清华大学出版社
北　京

内 容 简 介

《PHP 项目开发全程实录（第 3 版）》以九九度供求信息网、BCTY365 网上社区、办公自动化管理系统、网络在线考试系统、物流配送信息网、学校图书馆管理系统、博客管理系统、online 影视 365 网、明日知道网站和电子商务网站 10 个实际项目开发程序为案例，从软件工程的角度出发，按照项目的开发顺序，全面、系统地介绍了程序开发流程。从开发背景、需求分析、系统功能分析、数据库分析、数据库设计、网站开发到网站发布，对每一过程都进行了详细介绍。

本书及光盘特色还有：10 套项目开发完整案例，项目开发案例的同步视频和其源程序。登录网站还可获取各类资源库（模块库、题库、素材库）等项目案例常用资源，网站还提供技术论坛支持等。

本书案例涉及行业广泛，实用性非常强，通过本书的学习，读者既可以了解各个行业的特点，能够针对某一行业进行网站开发，也可以通过光盘中提供的案例源代码和数据库进行二次开发，以减少开发系统所需要的时间。

图书在版编目（CIP）数据

PHP 项目开发全程实录/辛洪郁，张鑫编著．—3 版．—北京：清华大学出版社，2013
（软件项目开发全程实录）

ISBN 978-7-302-33743-0

I．①P…　II．①辛…　②张…　III．①PHP 语言-程序设计　IV．TP312

中国版本图书馆 CIP 数据核字（2013）第 204535 号

责任编辑： 赵洛育
封面设计： 陈　敏
版式设计： 文森时代
责任校对： 李虎斌
责任印制：
出版发行： 清华大学出版社
　　网　　址： http://www.tup.com.cn，http://www.wqbook.com
　　地　　址： 北京清华大学学研大厦 A 座　　**邮　　编：** 100084
　　社 总 机： 010-62770175　　**邮　　购：** 010-62786544
　　投稿与读者服务： 010-62776969，c-service@tup.tsinghua.edu.cn
　　质量反馈： 010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：
装 订 者：
经　　销： 全国新华书店
开　　本： 185mm×260mm　**印　　张：** 28.25　**字　　数：** 790 千字
　（附 DVD 光盘 1 张）
版　　次： 2008 年 6 月第 1 版　2013 年 10 月第 3 版　**印　　次：** 2013 年 10 月第 1 次印刷
印　　数： 16001～20000
定　　价： 69.80 元

产品编号：051405-01

前言（第3版）

Preface 3rd Edition

编写目的与背景

众所周知，当前社会需求和高校课程设置严重脱节，一方面企业找不到可迅速上手的人才，另一方面大学生就业难。如果有一些面向工作应用的案例参考书，让大学生得以参考，并能亲手去做，势必能缓解这种矛盾。本书就是这样一本书：项目开发案例型的、面向工作应用的软件开发类图书。编写本书的首要目的就是架起让学生从学校走向社会的桥梁。

其次，本书以完成小型项目为目的，让学生切身感受到软件开发给工作带来实实在在的用处和方便，并非只是枯燥的语法和陌生的术语，从而激发学生学习软件的兴趣，让学生变被动学习为自主自发学习。

第三，本书的项目开发案例过程完整，不但适合在学习软件开发时作为小型项目开发的参考书，而且可以作为毕业设计的案例参考书。

第四，丛书第1版于2008年6月出版，于2011年1月改版，因为编写细腻，配备全程视频备受读者瞩目，丛书累计销售16万册，成为近年来最受欢迎的软件开发项目案例类丛书之一。

在以上背景下，我们根据读者朋友的反馈，与时俱进，对丛书进行了改版。

本书特点

视频讲解

对于初学者来说，视频讲解是最好的导师，它能够引导初学者快速入门，使初学者感受到编程的快乐和成就感，增强进一步学习的信心。鉴于此，本书为每一个案例都配备了视频讲解，初学者可以通过视频讲解实现案例中的功能。

典型案例

本书案例均从实际应用角度出发，应用了当前流行的技术，涉及的知识广泛，读者可以从每个案例中积累丰富的实战经验。

代码注释

为了便于读者阅读程序代码，书中的代码均提供了详细的注释，并且整齐地纵向排列，可使读者快速领略作者意图。

代码贴士

案例类书籍通常会包含大量的程序代码，冗长的代码往往令初学者望而生畏。为了方便读者阅读和理解代码，本书避免了连续大篇幅的代码，将其分割为多个部分，并对重要的变量、方法和知识点设计了独具特色的代码贴士。

知识扩展

为了增加读者的编程经验和技巧，书中每个案例都标记有注意、技巧等提示信息，并且在每章中都提供有一项专题技术。

本书约定

由于篇幅有限，本书每章并不能逐一介绍案例中的各模块。作者选择了基础和典型的模块进行介绍，对于功能重复的模块，由于技术、设计思路和实现过程基本雷同，因此没有在书中体现。读者在学习过程中若有相关疑问，请登录本书官方网站。本书中涉及的功能模块在光盘中都附带有视频录像，方便读者学习。

适合读者

本书适合作为计算机相关专业的大学生、软件开发相关求职者和爱好者的毕业设计和项目开发的参考书。

本书服务

为了给读者提供更为方便快捷的服务，读者可以登录本书官方网站：www.rjkflm.com，或者加入QQ：4006751066 进行交流。

本书作者

本书由明日科技软件开发团队组织编写，主要由辛洪郁、张鑫执笔，如下人员也参与了本书的编写工作，他们是：王雨竹、王小科、王国辉、杨丽、顾彦玲、赛奎春、高春艳、陈英、宋禹蒙、刘佳、刘莉莉、陈丹丹、隋光宇、郭鑫、刘志铭、李伟、张金辉、李慧、刘欣、李继业、潘凯华、赵永发、寇长梅、赵会东、王敬洁、李浩然、苗春义、刘清怀、张世辉、张领等，在此一并感谢！

在编写本书的过程中，我们本着科学、严谨的态度，力求精益求精，但错误、疏漏之处在所难免，敬请广大读者批评指正。

感谢您购买本书，希望本书能成为您的良师益友，成为您步入编程高手之路的踏脚石。

宝剑锋从磨砺出，梅花香自苦寒来。祝读书快乐！

编　　者

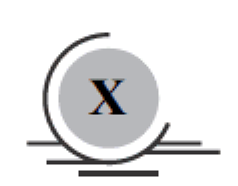


目　录

Contents






X

第 1 章

九九度供求信息网

（Apache+PHP+phpMyAdmin+MySQL 5.0 实现）

在全球知识经济和信息化高速发展的今天，信息化是决定企业成败的关键因素，企业需要在网站上发布供求信息，以促使企业在同领域中得到突飞猛进的发展。

一个广泛的、快速的、自由的信息交流平台，为用户带来方便的同时，也会给企业带来无限商机。于是，以互联网为基础的信息交流平台，即九九度供求信息网出现了。九九度供求信息网致力于优化信息交流，实现信息的快速交流。通过阅读本章，可以学习到：

- 使当前窗口承载框架页中的超链接页面
- 如何自动计算以系统日期为基数的相对日期
- do…While 循环语句的应用
- 查询关键字描红技术
- 在 Windows 操作系统下搭建 PHP 开发环境
- 在 Windows 操作系统下创建数据库和数据表
- PowerDesigner 建模的应用
- 单元测试技术
- 框架技术在 Web 网站中的应用
- 如何发布网站
- 表单数据的两种提交方式
- 应用 phpMyAdmin 工具创建和删除数据库、数据表

1.1 开 发 背 景

×××信息科技有限公司是一家以整合渠道资源为主的高科技公司。为了扩大企业规模，增强企业的竞争力，该公司决定向多元化发展，借助 Internet 在国内的快速发展，聚集部分资金投入网站建设，为企业和用户提供综合信息服务，以向企业提供有偿信息服务为盈利方式，打造一个全新的供求信息网。例如，提供企业广告、发布各类免费供求信息、发布企业付费信息等服务方式。现需要委托其他单位开发一个综合信息网站。

1.2 系 统 分 析

1.2.1 需求分析

对于信息网站来说，用户的访问量是至关重要的。如果网站的访问量很低，那么就很少有企业会要求为他提供有偿服务，也就没有利润可言了。因此，信息网站必须为用户提供大量的、免费的、有价值的信息才能够吸引用户。为此，网站不仅要为企业提供各种有偿服务，还需要额外为用户提供大量的无偿服务。通过与企业的实际接触和沟通，确定网站应包括招聘信息、求职信息、培训信息、公寓信息、家教信息、车辆信息、物品求购、物品出售、求兑出兑、寻求合作、企业广告等服务。

通过实际调查，要求供求信息网具有以下功能：

☑ 界面设计美观大方、方便、快捷、操作灵活，树立企业形象。
☑ 实现强大的供求信息查询，支持模糊查询。
☑ 用户不需要注册，便可免费发布供求信息。
☑ 免费发布的供求信息必须经后台审核后才能正式发布，避免不良信息。
☑ 支持海量数据录入。
☑ 由于供求信息数据量大，后台应该可以随时清理数据。

1.2.2 可行性分析

根据《GB8567－88 计算机软件产品开发文件编制指南》中可行性分析的要求，制定可行性研究报告如下。

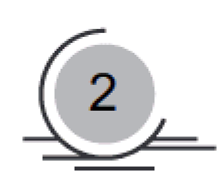
1．引言

（1）编写目的

为了给企业的决策层提供是否进行项目实施的参考依据，现以文件的形式分析项目的风险、项目需要的投资与效益。

（2）背景

×××信息科技有限公司是一家以整合渠道资源为主的高科技公司。企业为了不断满足客户的需求，为达到企业在同行业领域中的领先地位，现需要委托其他公司开发一个综合信息网，项目名称为九九度供求信息网。





2．可行性研究的前提

（1）要求

九九度供求信息网要求能够提供信息搜索、信息定位描红、发布免费信息、发布付费信息、发布企业广告和对各类发布的信息进行审核、删除、检索等功能。

（2）目标

九九度供求信息网的主要目标是提供强大的搜索功能、准确的信息描红定位功能、付费信息的管理功能、免费信息的审核和删除功能。

（3）条件、假定和限制

项目需要在两个月内交付用户使用。系统分析师需要 3 天内到位，用户需要 4 天时间确认需求分析文档。去除员工两个月的正常休息日 16 天，那么程序开发人员需要在 1 个月零几天的时间内进行系统设计、程序编码、系统测试、程序调试和网站部署工作。

（4）评价尺度

根据用户的要求，系统应以搜索引擎为主，对于发布的供求信息应能及时准确地保存、审核、查询、描红定位。由于用户存在多个营业点，系统应具有局域网操作的能力，在多个营业点同时运行系统时，系统中各项操作的延时不能超过 10 秒钟。此外，在系统出现故障时，应能及时进行恢复。

3．投资及效益分析

（1）支出

根据系统的规模及两个月的项目开发周期，公司决定投入 5 个人。因此，公司将直接支付 8 万元的工资及各种福利待遇。在项目安装及调试阶段，用户培训、员工出差等费用支出需要 2 万元。在项目维护阶段预计需要投入 2 万元的资金，累计项目投入需要 12 万元资金。

（2）收益

用户提供项目资金 30 万元。对于项目运行后进行的改动，采取协商的原则根据改动规模额外提供资金。因此从投资与收益的效益比上，公司可以获得 18 万元的利润。

项目完成后，将给公司提供资源储备，包括技术、经验的积累，以后再开发类似的项目时，可以极大地缩短项目开发周期。

4．结论

根据上面的分析，技术上不会存在问题，因此项目延期的可能性很小；在效益上，公司投入 5 个人、两个月的时间获利 18 万元，比较可观；在公司今后的发展上，可以储备网站开发的经验和资源，因此认为该项目可以开发。

1.2.3　编写项目计划书

根据《GB8567－88 计算机软件产品开发文件编制指南》中的项目开发计划要求，结合单位实际情况，设计项目计划书如下。

1．引言

（1）编写目的

为了保证项目开发人员按时保质地完成预订目标，更好地了解项目实际情况，按照合理的顺序开

展工作，现以书面的形式将项目开发生命周期中的项目任务范围、项目团队组织结构、团队成员的工作责任、团队内外沟通协作方式、开发进度、检查项目工作等内容描述出来，作为项目相关人员之间的共识和约定以及项目生命周期内的所有项目活动的行动基础。

（2）背景

九九度供求信息网是本公司与×××信息科技有限公司签定的待开发项目，网站性质为信息服务类型，可为信息发布者有偿或无偿提供招聘、求职、培训、求购、公寓、车辆、房屋和出售等信息。项目周期为两个月，项目背景规划如表 1.1 所示。

表 1.1　项目背景规划

项 目 名 称	签定项目单位	项目负责人	项目承担部门
九九度供求信息网	甲方：×××信息科技有限公司	甲方：赵经理	设计部门 开发部门 测试部门
	乙方：×××网络科技有限公司	乙方：张经理	

2．概述

（1）项目目标

项目目标应当符合 SMART 原则，把项目要完成的工作用清晰的语言描述出来。九九度供求信息网的项目目标如下：

九九度供求信息网主要用来为用户提供信息服务，对于生活和工作中的各类信息都应尽可能地全部包括在内，例如公寓、求职、招聘、培训、招商、房屋、车辆、出售、求购等信息。项目发布后，要实现能够为用户生活和工作带来极大的方便并提高企业知名度、为企业产品宣传节约大量成本的目标。整个项目需要在两个月的期限结束后，交给客户进行验收。

（2）产品目标与范围

一方面九九度供求信息网能够为企业节省大量人力资源，企业不再需要大量的业务人员去跑市场，间接为企业节约了成本。另一方面，九九度供求信息网能够收集海量供求信息，将会有大量用户访问网站，有助于提高企业知名度。

（3）应交付成果

项目开发完成后，交付的内容如下：

☑ 以光盘的形式交付九九度供求信息网的源程序、网站数据库文件、系统使用说明书。
☑ 客户方应用自己的服务器，因此需要乙方架设 Apache 服务器、安装 PHP 开发环境、协助甲方购买域名，将开发的九九度供求信息网发布到互联网上运行。
☑ 网站发布到互联网上以后，进行后期的 6 个月无偿维护与服务，超过 6 个月后进行网站有偿维护与服务。

（4）项目验收方式与依据

项目验收分为内部验收和外部验收两种方式。在项目开发完成后，首先进行内部验收，由系统测试员根据用户需求和项目目标进行验收。项目在通过内部验收后交给用户进行验收，验收的主要依据为需求规格说明书。

3．项目团队组织

（1）组织结构

为了完成九九度供求信息网的项目开发，公司组建了一个临时的项目团队，由项目经理、系统分



析师、PHP 开发工程师、网页设计师和系统测试员构成，如图 1.1 所示。

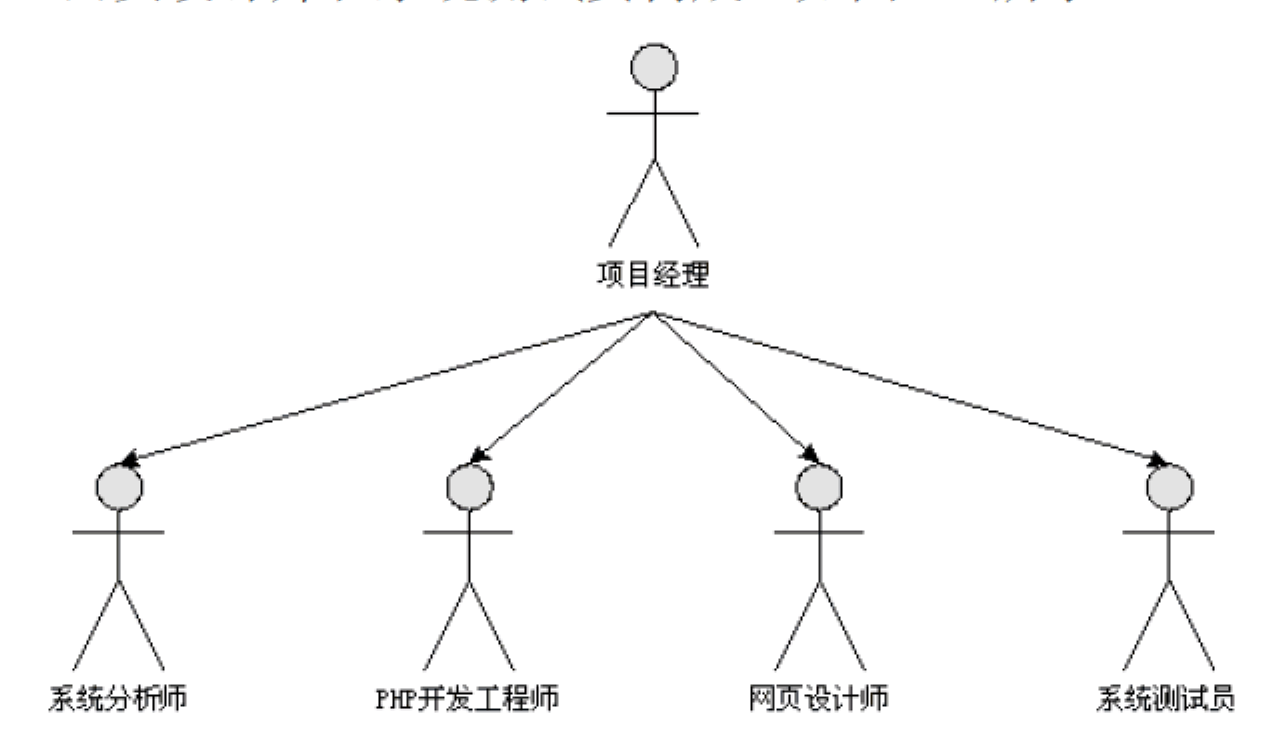


图 1.1　项目团队组织结构图

（2）人员分工

为了明确项目团队中每个人的任务分工，现制定人员分工表，如表 1.2 所示。

表 1.2　人员分工表

姓　名	技 术 水 平	所 属 部 门	角　色	工 作 描 述
张达明	MBA	项目开发部	项目经理	负责项目的审批、决策的实施、项目的前期分析、策划、项目开发进度的跟踪、项目质量的检查
周兴伟	高级系统分析师	项目开发部	系统分析师	负责系统功能分析、系统框架设计
邹紫璇	高级 PHP 工程师	项目开发部	PHP 开发工程师	负责软件前后台设计与编码
王惠子	高级美工设计师	设计部	网页设计师	负责网页风格的确定、网页图片的设计
舒心怡	高级系统测试工程师	项目开发部	系统测试员	对软件进行测试、编写软件测试文档

1.3　系 统 设 计

1.3.1　系统目标

根据需求分析的描述以及与用户的沟通，现制定网站实现如下目标：

- ☑ 系统采用人机对话方式，界面简洁、框架清晰、美观大方。
- ☑ 灵活快速地填写供求信息，使信息传递更快捷。
- ☑ 信息查询灵活、方便，数据存储安全可靠。
- ☑ 实现强大的后台审核功能。
- ☑ 实现强大的搜索引擎，支持模糊查询、关键字描红功能等。
- ☑ 对用户输入的数据，系统进行严格的数据检验，尽可能排除人为的错误。
- ☑ 网站最大限度地实现易维护性和易操作性。
- ☑ 为充分展现网站的交互性，供求信息网采用动态网页技术实现用户信息在线发布。
- ☑ 具备完善的后台管理功能，能够及时、准确地对网站进行维护和更新。

1.3.2 系统功能结构

九九度供求信息网前台功能结构图如图 1.2 所示。

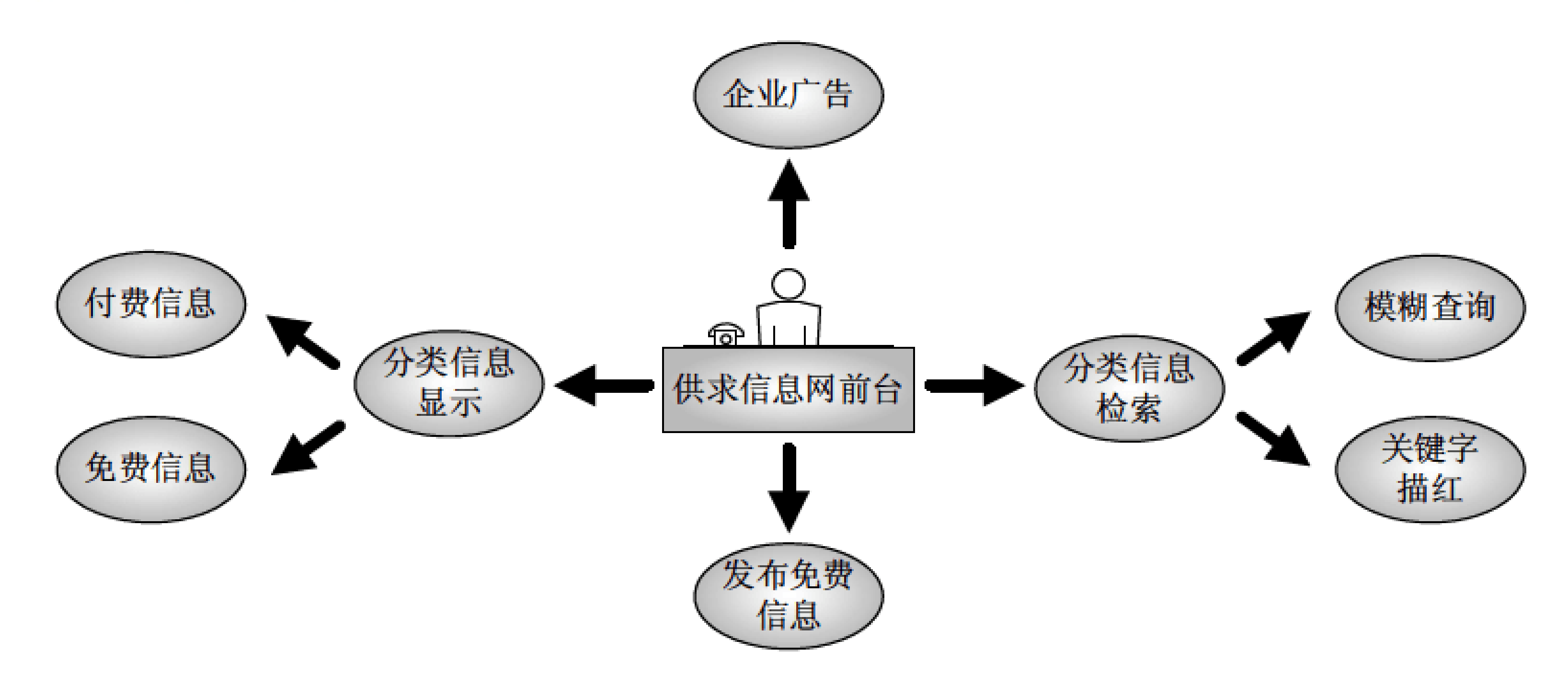


图 1.2 九九度供求信息网前台功能结构图

九九度供求信息网后台功能结构图如图 1.3 所示。

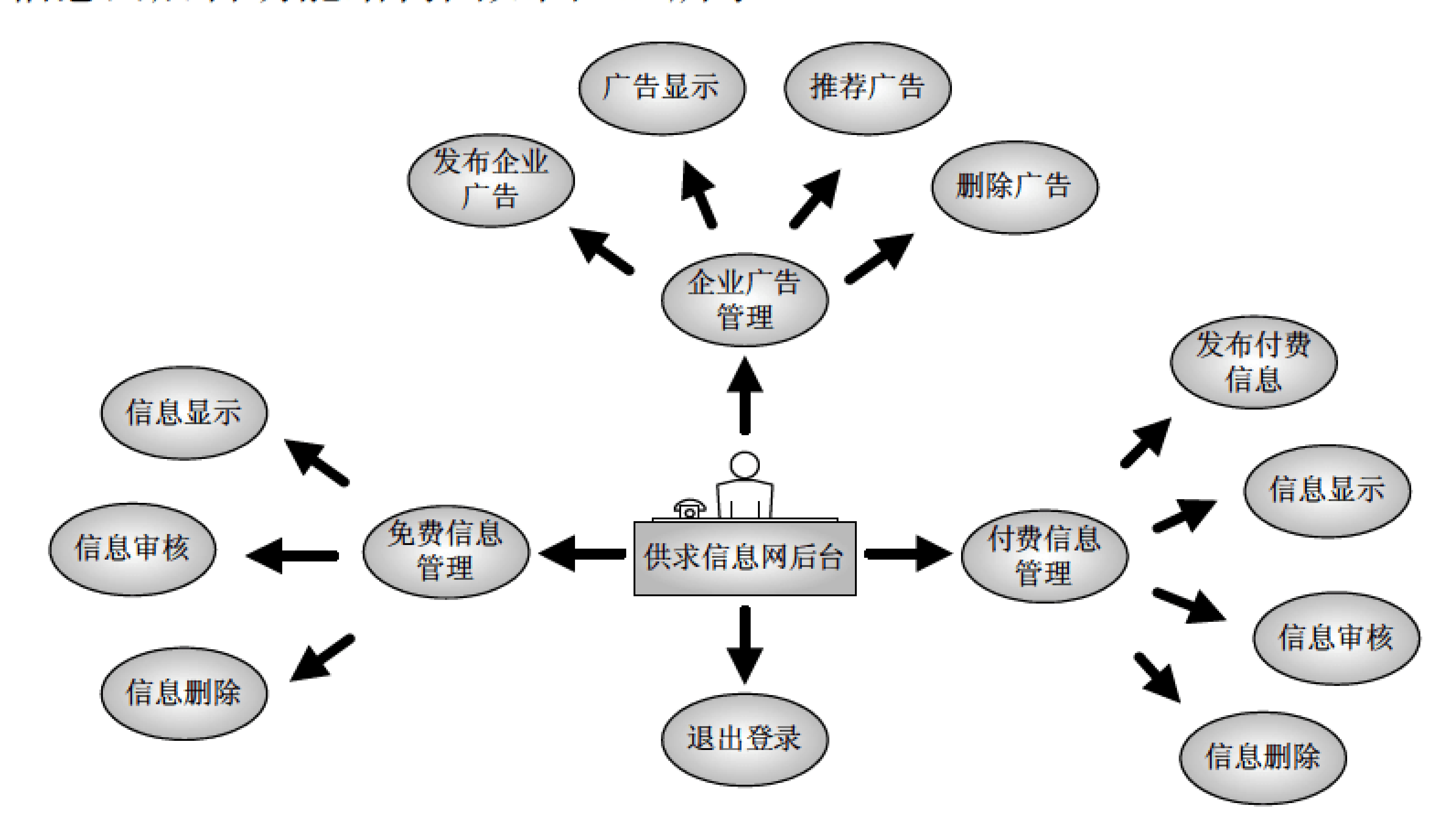


图 1.3 九九度供求信息网后台功能结构图

1.3.3 系统流程图

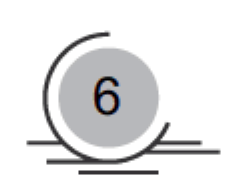
九九度供求信息网的系统流程如图 1.4 所示。

1.3.4 系统预览

九九度供求信息网由多个程序页面组成。下面列出几个典型页面，其他页面参见光盘中的源程序。

前台首页如图 1.5 所示，该页面用于实现各类信息的查询、企业广告信息显示、后台登录入口等功能。搜索引擎页面如图 1.6 所示，该页面用于实现各类信息的快速检索、查询关键字描红等功能。

发布免费信息页面如图 1.7 所示，该页面用于实现发布分类的免费信息功能。付费信息管理页面如图 1.8 所示，该页面用于实现付费信息分类查看、付费信息审核、付费信息删除等功能。





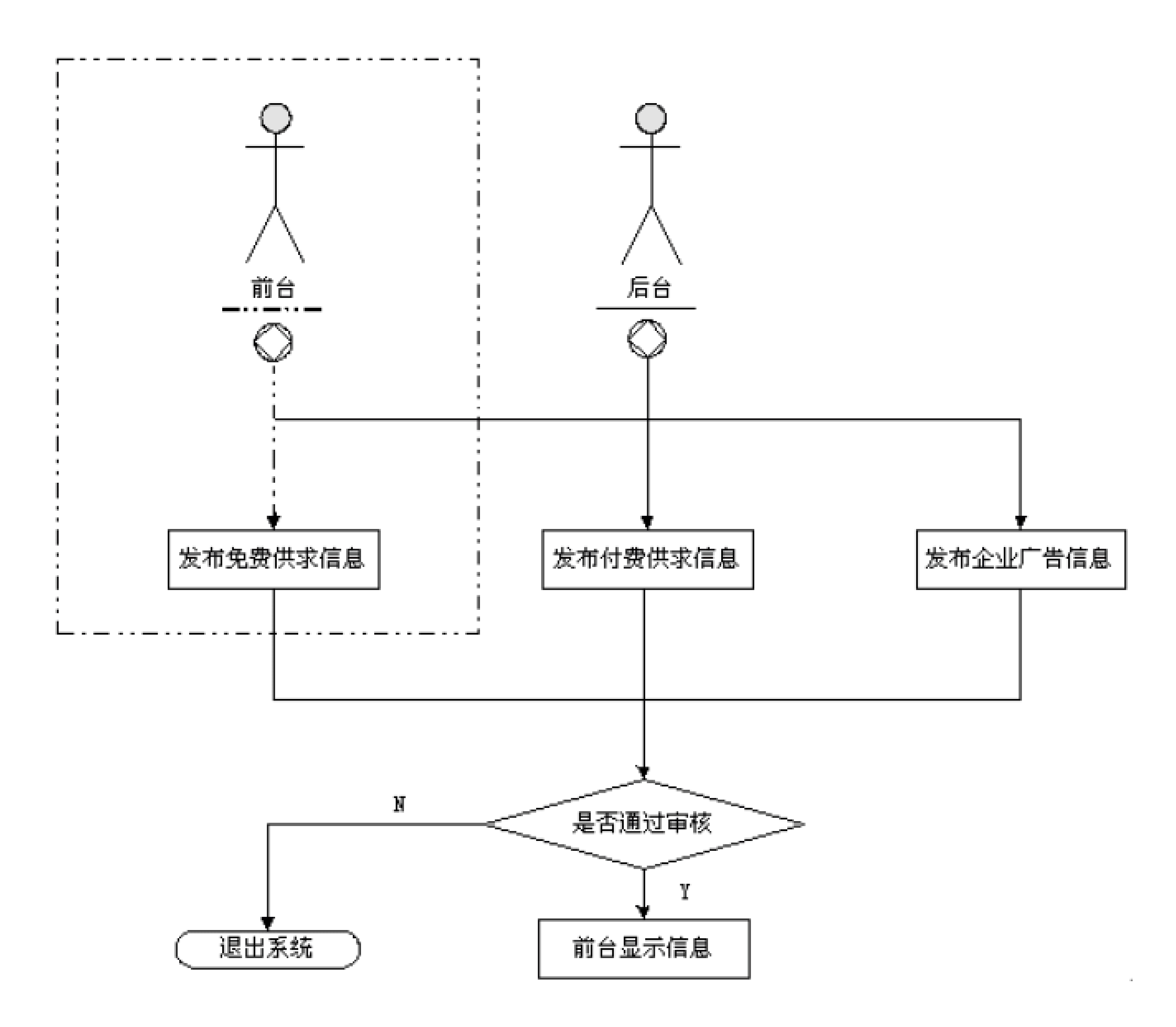


图 1.4　系统流程图



图 1.5　前台首页（光盘\TM\01\99pursey\index.php）

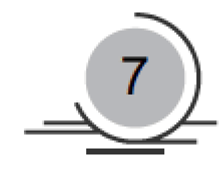
图 1.6　搜索引擎（光盘\TM\01\99pursey\admin\findinfo.php）



图 1.7　发布免费信息（光盘\TM\01\99pursey\release.php）



图 1.8　付费信息管理（光盘\TM\01\99pursey\admin\find_fufei.php）



免费信息管理页面如图 1.9 所示，该页面用于实现免费信息分类查看、免费信息审核、免费信息删除等功能。管理员登录页面如图 1.10 所示，该页面用于实现对管理员登录的用户名和密码进行验证等功能。





图 1.9　免费信息管理（光盘\TM\01\99pursey\admin\finddd.php）　图 1.10　管理员登录（光盘\TM\01\99pursey\admin\login.php）

1.3.5　开发环境

在开发九九度供求信息网时，该项目使用的软件开发环境如下。

1. 服务器端

☑　操作系统：Windows Server 2003/Linux（推荐）。
☑　服务器：Apache 2.2.8。
☑　PHP 软件：PHP 5.2.6。
☑　数据库：MySQL 5.0.51。
☑　MySQL 图形化管理软件：phpMyAdmin-2.10.3。
☑　开发工具：Dreamweaver 8。
☑　浏览器：IE 6.0 及以上版本。
☑　分辨率：最佳效果为 1024×768 像素。

2. 客户端

☑　浏览器：IE 6.0 及以上版本。
☑　分辨率：最佳效果为 1024×768 像素。

1.3.6　文件夹组织结构

在编写代码之前，可以把系统中可能用到的文件夹先创建出来（例如，创建一个名为 admin 的文件夹，用于保存网站的后台文件），这样不但可以方便以后的开发工作，也可以规范网站的整体架构。笔者在开发九九度供求信息网时，设计了如图 1.11 所示的文件夹

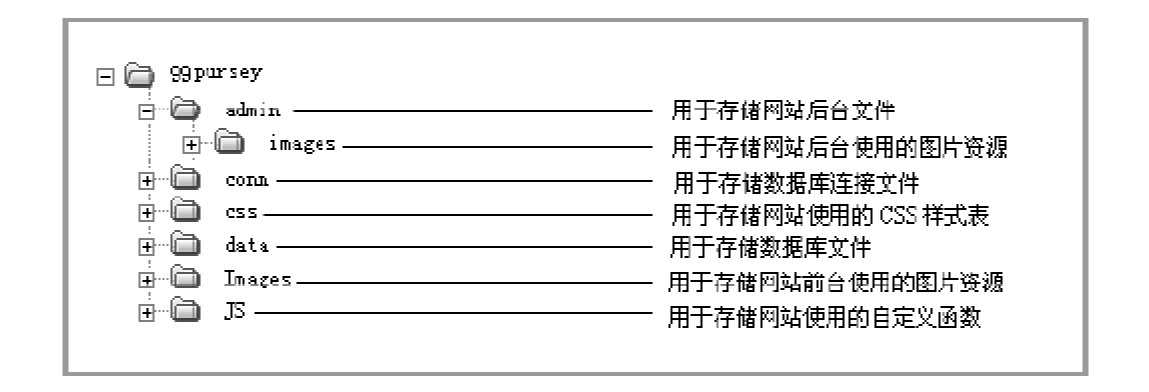
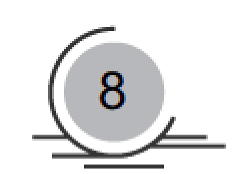


图 1.11　文件夹组织结构





组织结构图。在开发时，只需要将所创建的文件保存在相应的文件夹中即可。

1.4　在 Windows 操作系统下搭建 PHP 开发环境

PHP 能否高效、稳定地运行依赖于服务器的编译和执行，本节主要介绍如何在微软的 Windows 操作系统中架设安全、可靠的 PHP 运行环境。

1.4.1　在 Windows 下应用 AppServ 快速配置 PHP 开发环境

AppServ 是 PHP 网页架站工具组合包，可以将网络上免费的架站资源重新包装成单一的安装程序。它提供了简易、快速的 PHP 运行环境的搭建机制，读者只需按照普通应用软件的安装方式就可以完成 Apache+MySQL+PHP+phpMyAdmin 的安装与配置工作。

下面以 AppServ-win32-2.5.10 为例来介绍 AppServ 的安装和使用方法。

安装 AppServ 之前应从官方网站 http://www.appservnetwork.com 下载最新版本的 AppServ-win32-2.5.10.exe 安装程序。

视频录像：TM\01\lx\在 Windows 下应用 AppServ 快速配置 PHP 开发环境.swf

在 Windows 操作系统下应用 AppServ 快速配置 PHP 开发环境的操作步骤如下：

（1）双击 AppServ-win32-2.5.10.exe 文件，打开如图 1.12 所示的 AppServ 启动窗口。

（2）单击图 1.12 中的 Next 按钮，打开如图 1.13 所示的 AppServ 安装协议窗口。

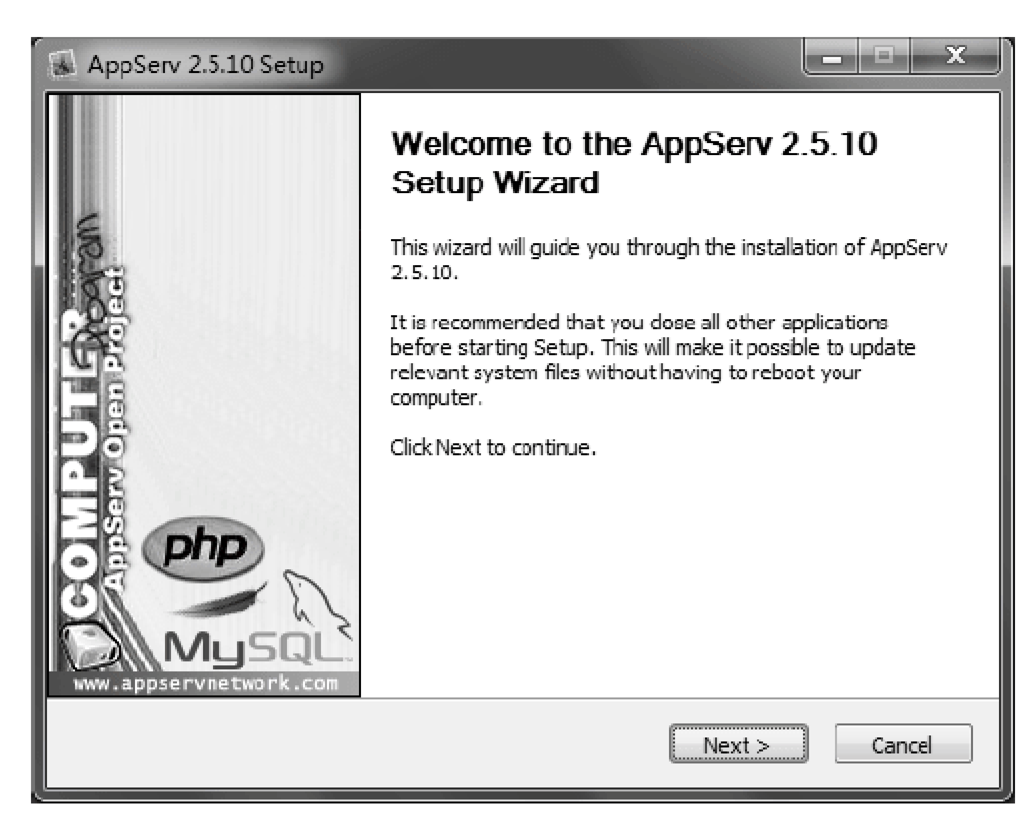



图 1.12　AppServ 启动窗口

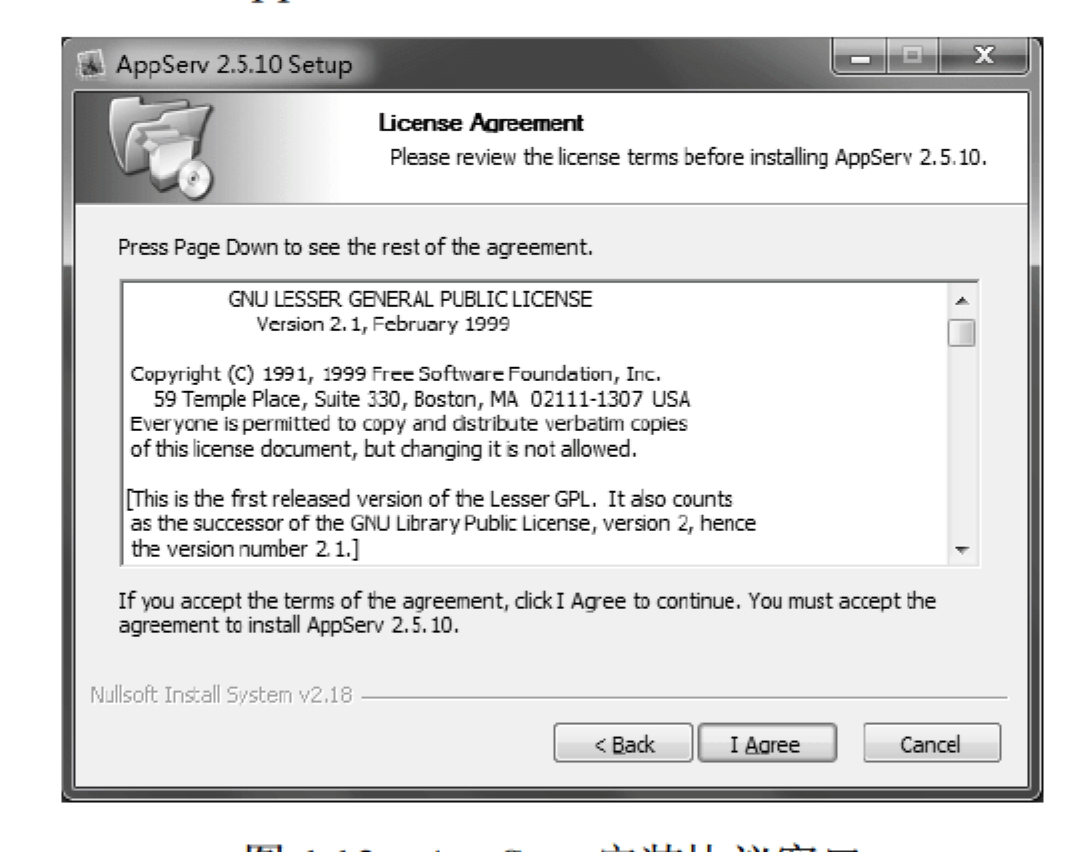


图 1.13　AppServ 安装协议窗口

（3）单击图 1.13 中的 I Agree 按钮，打开如图 1.14 所示的窗口，在该窗口中可以设置 AppServ 的安装路径（默认安装路径一般为 C:\AppServ）。AppServ 安装完成后，Apache、MySQL、PHP 都将以子目录的形式存储到该目录下。

（4）单击图 1.14 中的 Next 按钮，打开如图 1.15 所示的窗口，在该窗口中可以选择要安装的程序和组件（默认为全选状态）。

（5）在图 1.15 中单击 Next 按钮，打开如图 1.16 所示的窗口，该窗口主要用于设置 Apache 的端口号。

（6）单击图 1.16 中的 Next 按钮，打开如图 1.17 所示的窗口。该窗口主要对 MySQL 数据库的

root 用户的登录密码及字符集进行设置，这里将字符集设置为 GB2312 Simplified Chinese，表示 MySQL 数据库的字符集将采用简体中文形式。

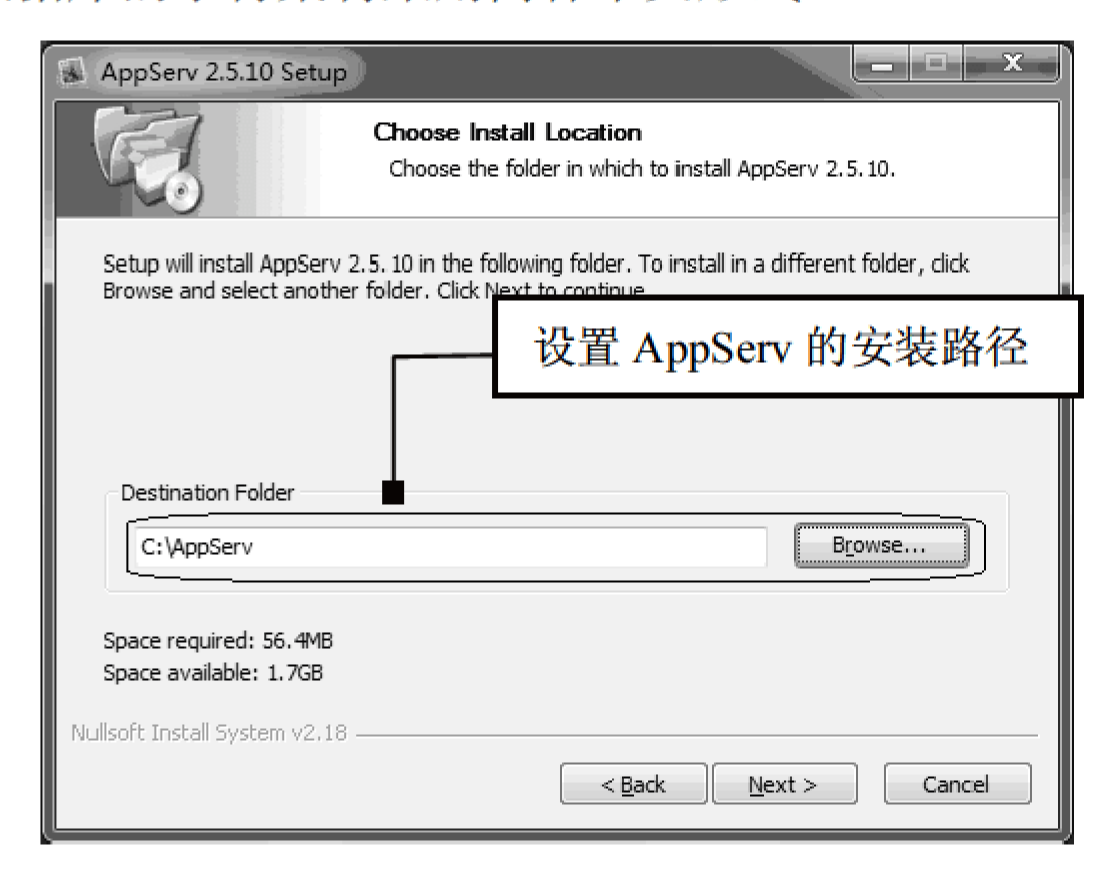


图 1.14　AppServ 安装路径选择

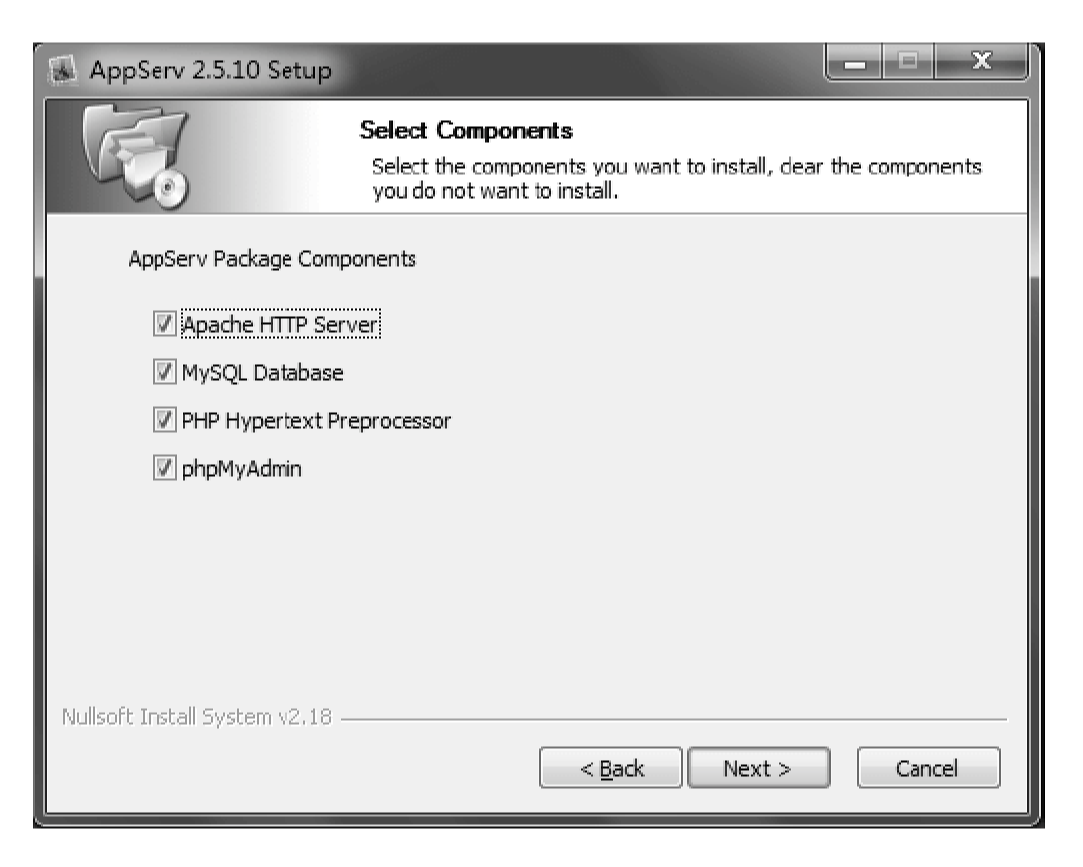


图 1.15　AppServ 安装选项

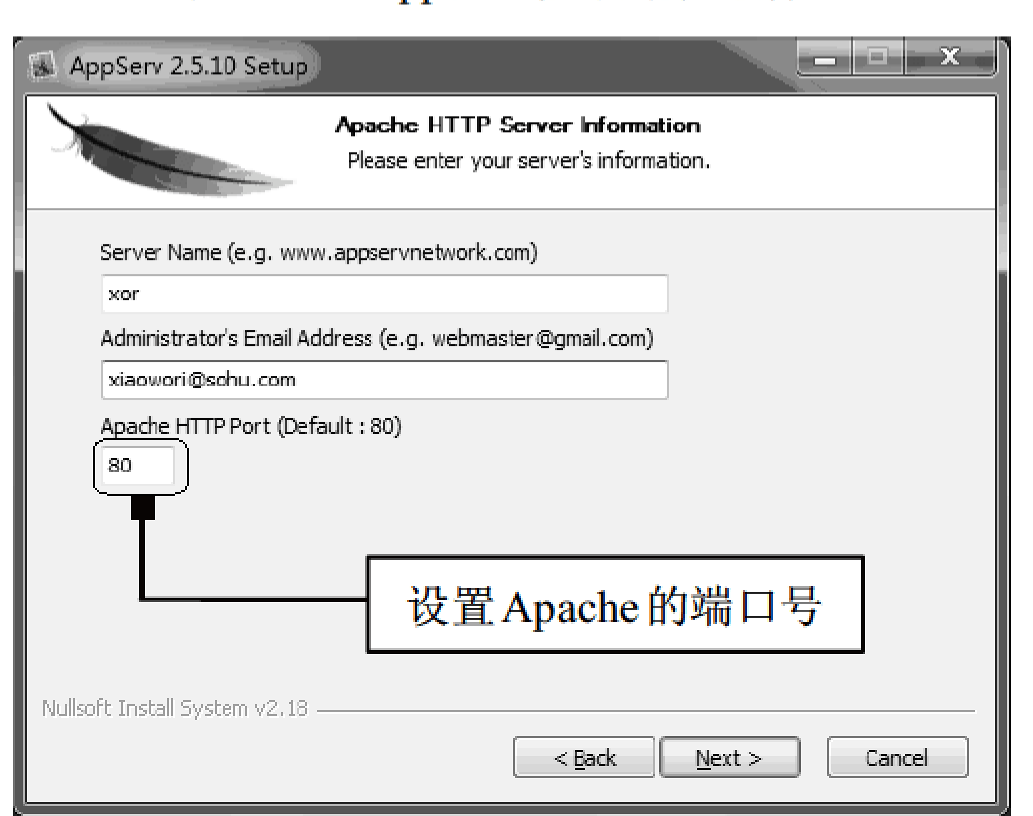


图 1.16　Apache 端口号设置



图 1.17　MySQL 设置

（7）单击图 1.17 中的 Install 按钮后开始安装，如图 1.18 所示。

（8）安装完成后可以选择在“开始”菜单的 AppServ 相关操作列表中启动 Apache 及 MySQL 服务，如图 1.19 所示。



图 1.18　AppServ 安装窗口

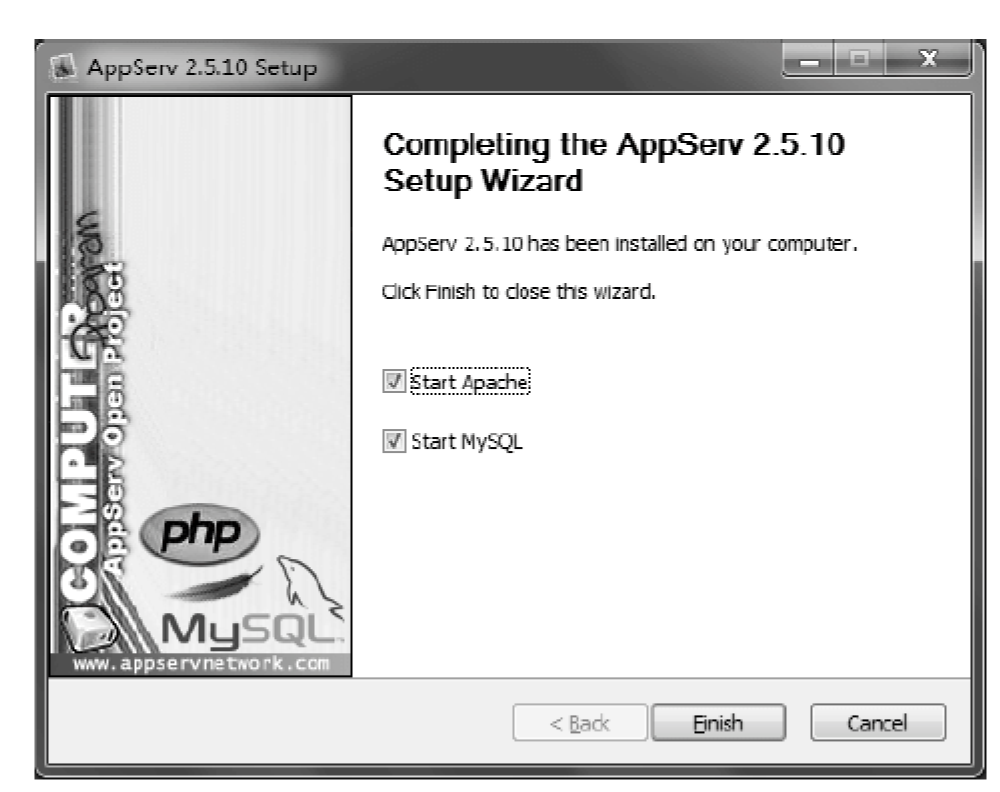
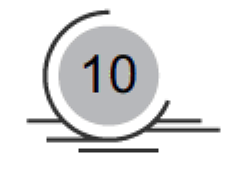


图 1.19　AppServ 安装完成窗口



（9）安装好 AppServ 后，整个目录默认安装在 C:\AppServ，此目录下包含 4 个子目录，如图 1.20 所示，用户可以将所有网页文件存放到 www 目录下。

（10）打开浏览器，在地址栏中输入 http://localhost 或者 http://127.0.0.1，如果打开如图 1.21 所示的网页，则说明 AppServ 安装成功。

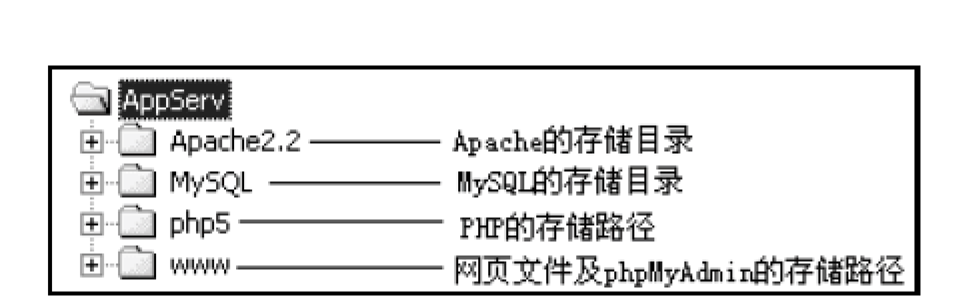


图 1.20　AppServ 目录结构

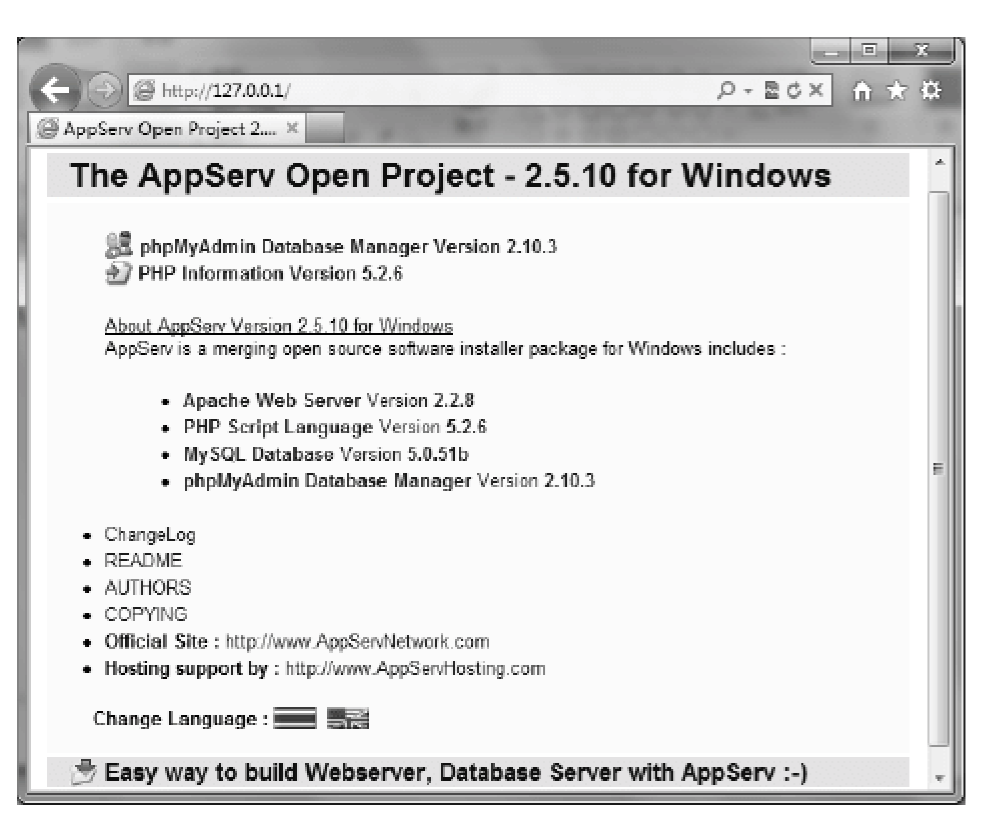


图 1.21　AppServ 测试页

1.4.2　Windows 下 Apache 的安装配置

Apache 是全世界使用范围最广的 Web 服务软件，超过 50%的网站都在使用 Apache 服务器，它以高效、稳定、安全、免费而成为了最受欢迎的服务器软件。

本节主要介绍如何在 Windows 操作系统中安装和配置 Apache 服务器。安装 Apache 服务器前，应到官方网站 http://www.apache.org 下载 Apache 的安装程序。

视频录像：TM\01\lx\Windows 下 Apache 的安装配置.swf

在 Windows 操作系统下实现 Apache 的安装配置的操作步骤如下：

（1）下载 Apache 的安装包 apache_2.0.59-win32-x86-no_ssl.msi 后，双击该安装包，打开如图 1.22 所示的 Apache 安装窗口。

（2）在图 1.22 中单击 Next 按钮，打开如图 1.23 所示的 Apache 许可协议窗口，选中 I accept the terms in the license agreement 单选按钮。

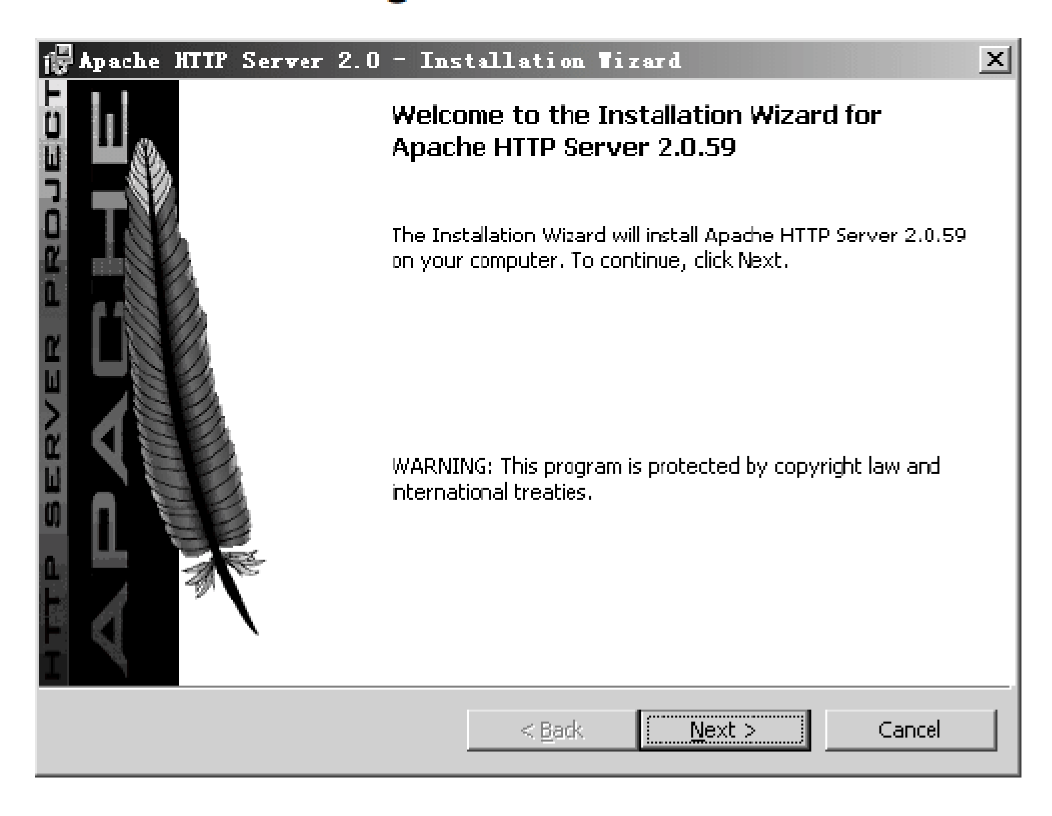


图 1.22　Apache 安装窗口

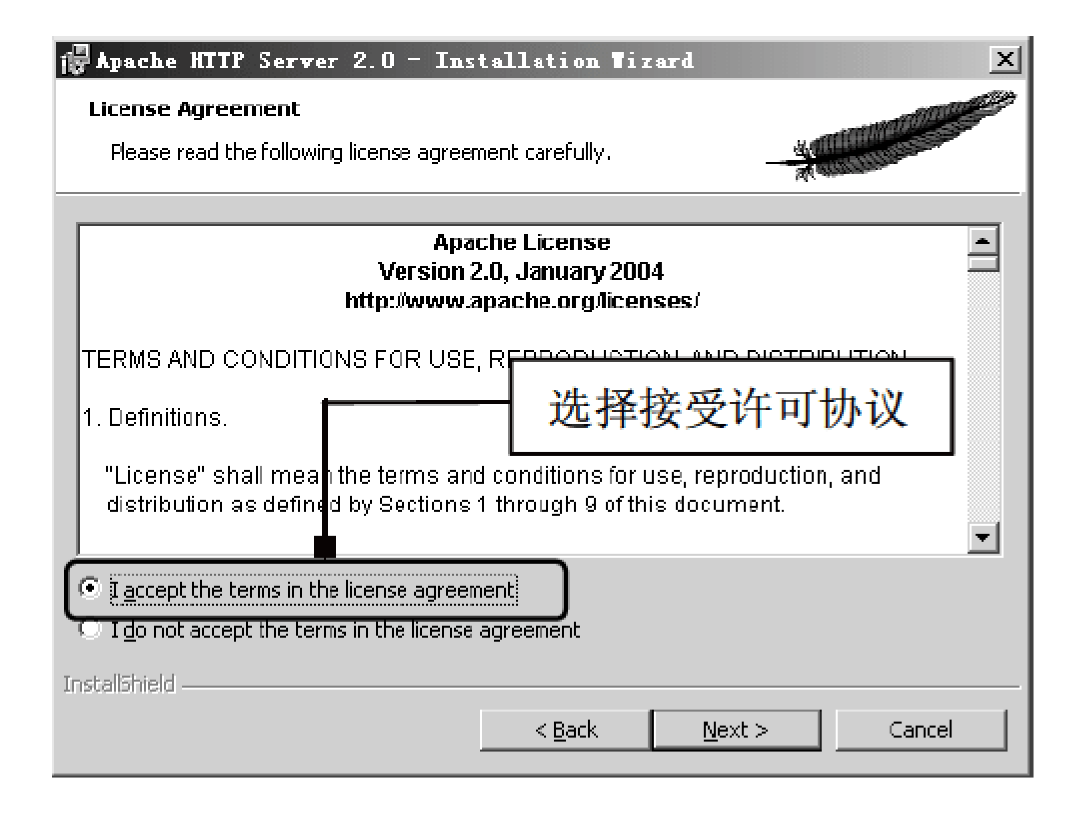
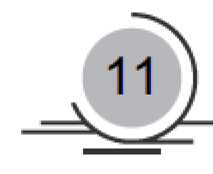


图 1.23　Apache 许可协议窗口



（3）在图 1.23 中单击 Next 按钮，打开如图 1.24 所示的 HTTP 服务窗口。

（4）在图 1.24 中单击 Next 按钮，打开如图 1.25 所示的输入服务器信息窗口。输入服务器的相关信息，如网络域名、服务器名和管理员邮箱等，这里可以根据用户的实际情况输入。在下方的单选按钮组中，如果选择第一项，则可以对任何用户开放 Apache 服务，同时设置服务器的侦听端口为 80；如果选择第二项，则只有本地用户可以连接和使用 Apache 服务。

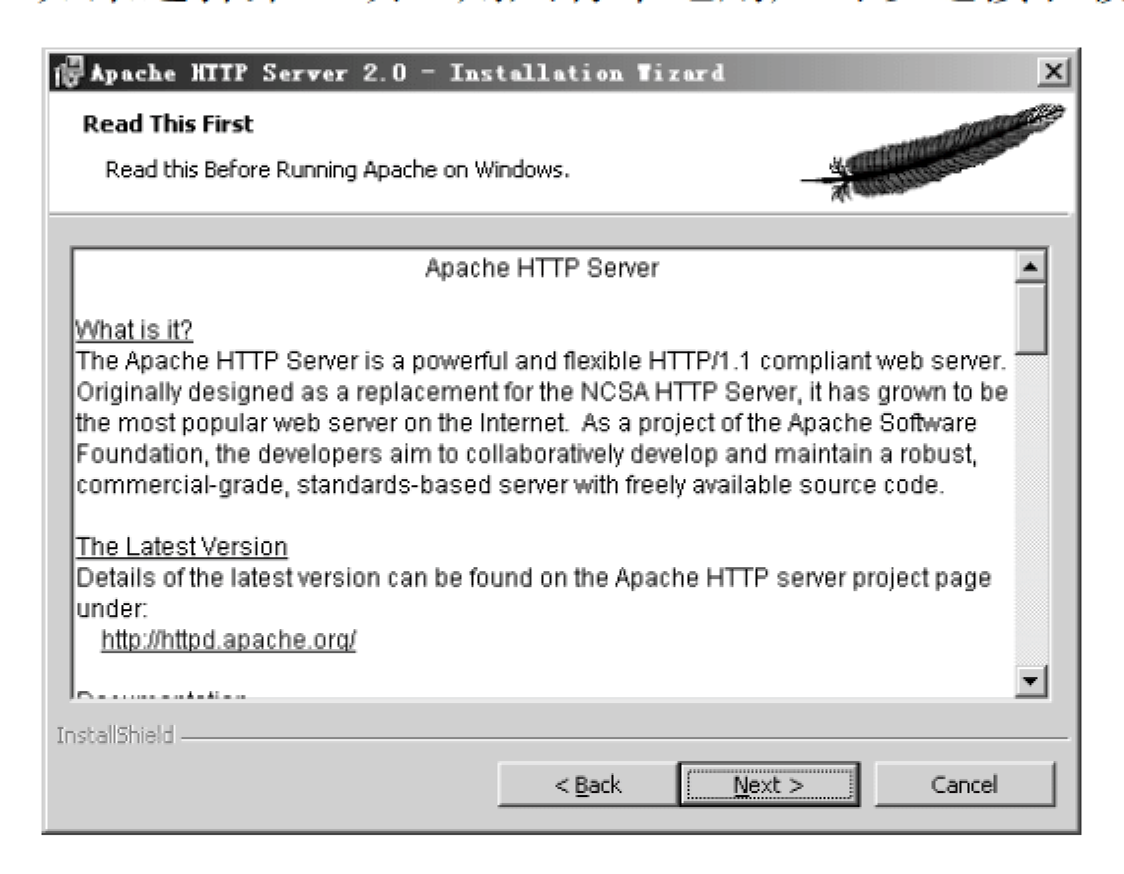


图 1.24　HTTP 服务窗口

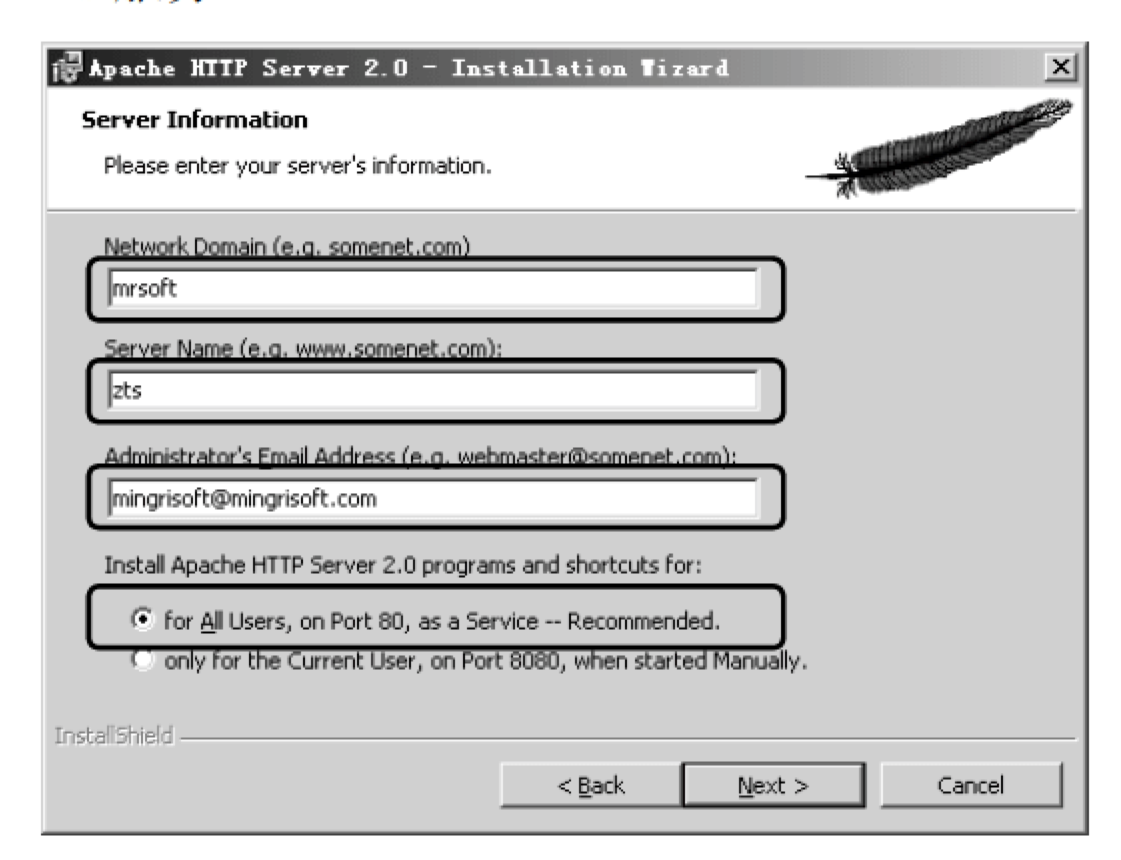


图 1.25　输入服务器信息窗口

注意　如果选择的是第一项，又同时安装了 IIS，那就必须修改 IIS 的默认端口，否则将导致 Apache 无法正常工作。更改 IIS 的默认侦听端口 80，可以在 IIS 的管理器中进行设置，或者停止 IIS 的服务也可以。

（5）设置完成后，单击图 1.25 中的 Next 按钮，打开如图 1.26 所示的选择安装方式窗口。这里有两种选择方式：Typical 典型安装和 Custom 自定义安装。通常，用户都选择典型安装方式，单击 Next 按钮，打开如图 1.27 所示的安装窗口。



图 1.26　选择安装方式窗口

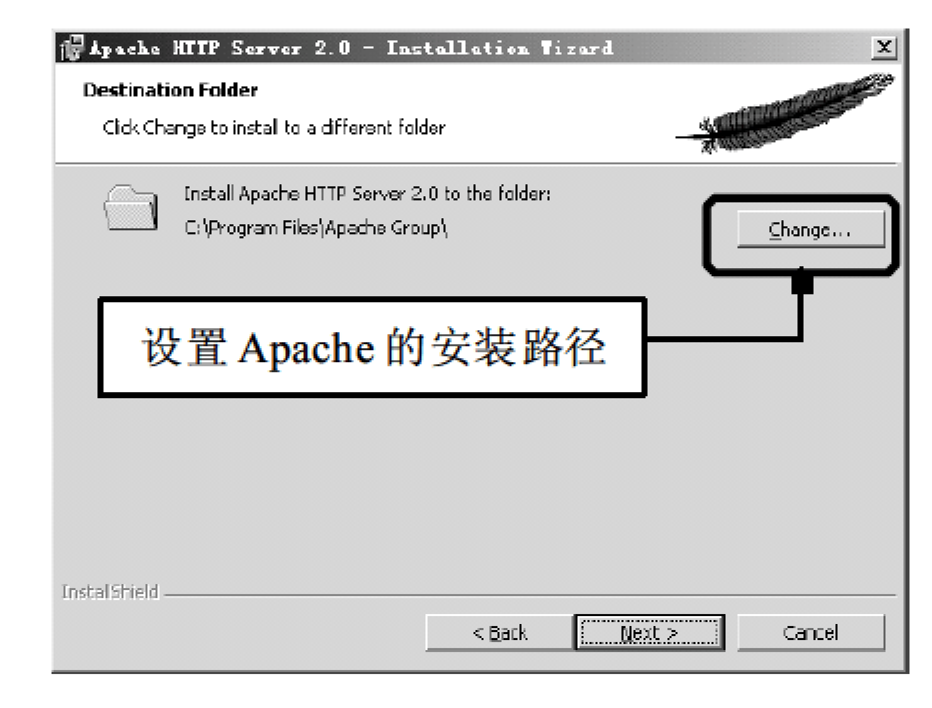


图 1.27　设置安装的路径

（6）在如图 1.27 所示的设置安装路径窗口中可以选择安装的路径，单击 Change 按钮，在打开的如图 1.28 所示的对话框中修改文件的安装路径。

（7）在图 1.28 中单击 OK 按钮，返回图 1.27，单击 Next 按钮，打开如图 1.29 所示的准备安装窗口。

（8）在图 1.29 中单击 Install 按钮，打开如图 1.30 所示的窗口开始安装。

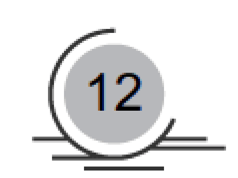








图 1.28　修改安装的路径

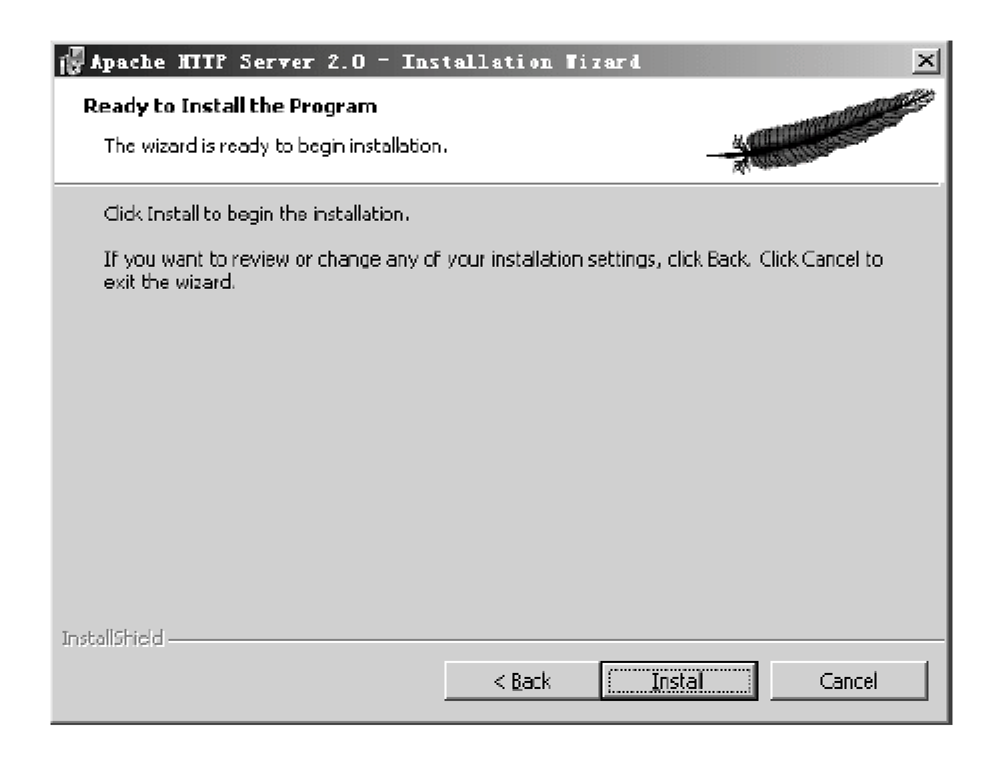
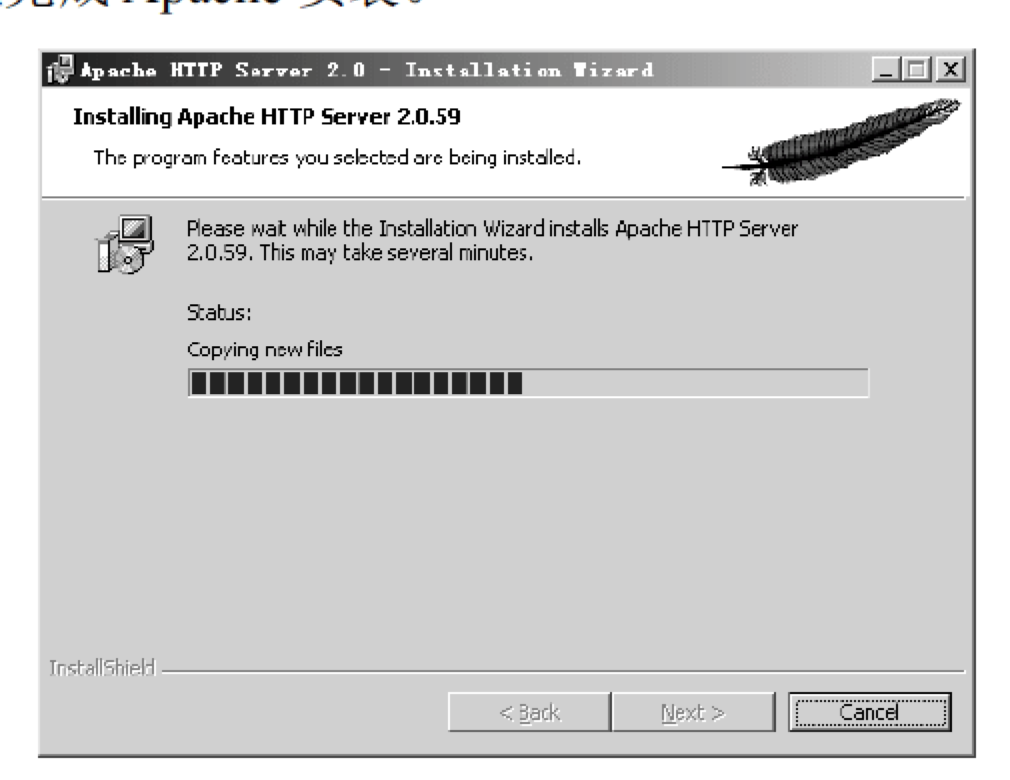


图 1.29　准备安装

（9）图 1.30 中所有文件复制完成后，Apache 安装完成，打开如图 1.31 所示的窗口，单击 Finish 按钮完成 Apache 安装。



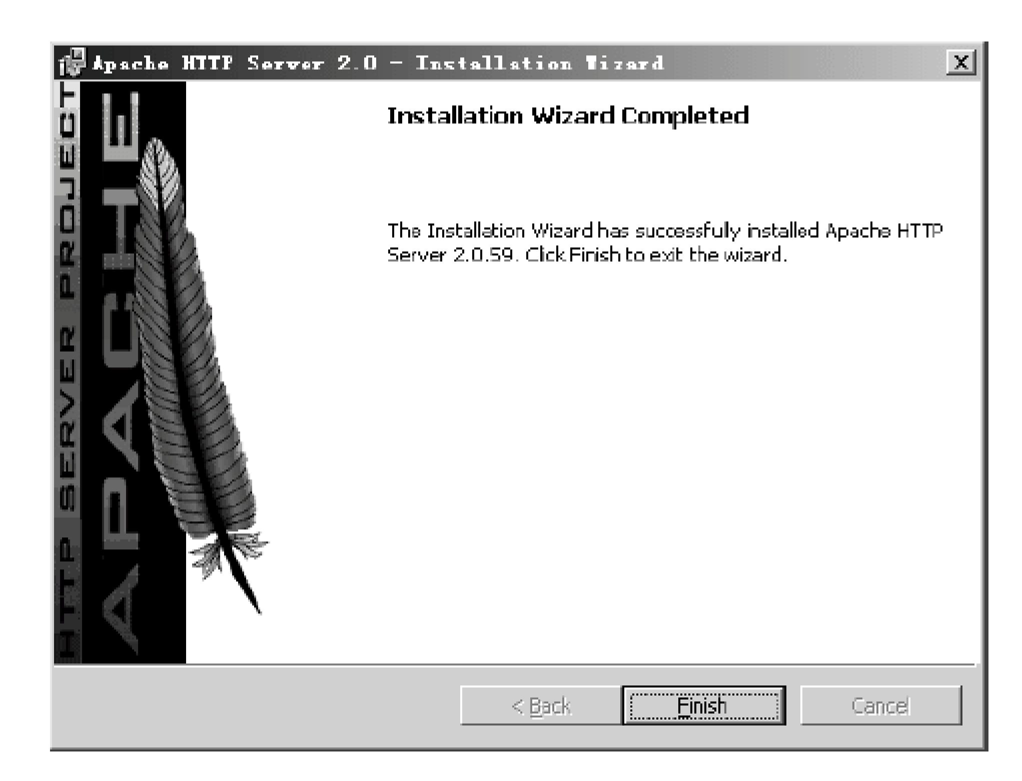
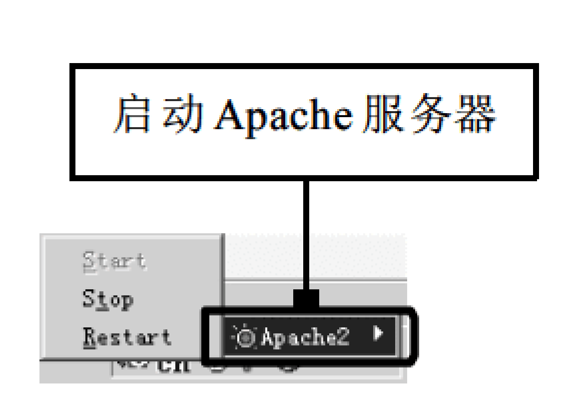


图 1.30　复制文件　　　　图 1.31　安装完成

（10）测试 Apache 服务是否安装成功。选择“开始”/“所有程序”命令，在弹出的菜单中能够看到 Apache 服务器的相关操作列表，同时如果在系统托盘中有一个图标，则表示 Apache 服务已经启动。

（11）单击图标后，打开如图 1.32 所示的窗口，在该窗口中，选择 Stop 选项，表示停止 Apache 服务器；选择 Start 选项，表示启动 Apache 服务器（Apache 服务器的当前状态为停止时，该功能可用）；选择 Restart 选项，表示重新启动 Apache 服务器。双击图标后，打开如图 1.33 所示的窗口，可以进行启动和停止服务器的各种操作。



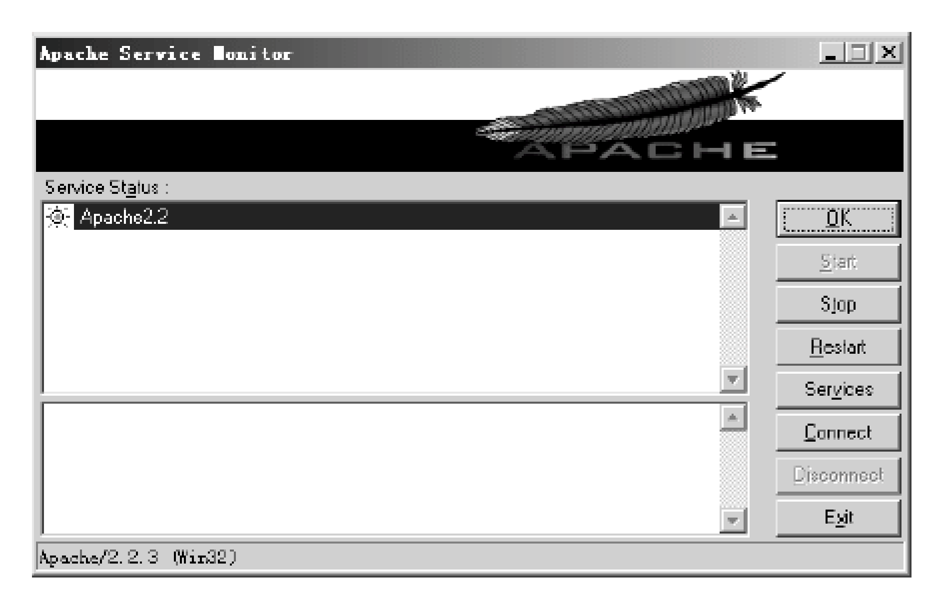
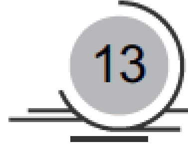


图 1.32　左键菜单　　　　图 1.33　Apache 服务器的监视器



（12）在浏览器中输入 http://localhost/或者输入 http://127.0.0.1/，如果能够浏览到如图 1.34 所示的窗口，说明 Apache 服务器安装成功。

（13）Apache 服务器安装成功后，接下来对 Apache 服务器进行配置，以便 Apache 服务器能够识别 PHP 文件。配置 Apache 服务器主要是在 Apache 安装目录下的 conf 子目录中的 httpd.conf 文件中进行的，找到该文件并用记事本等文本编辑器打开该文件。

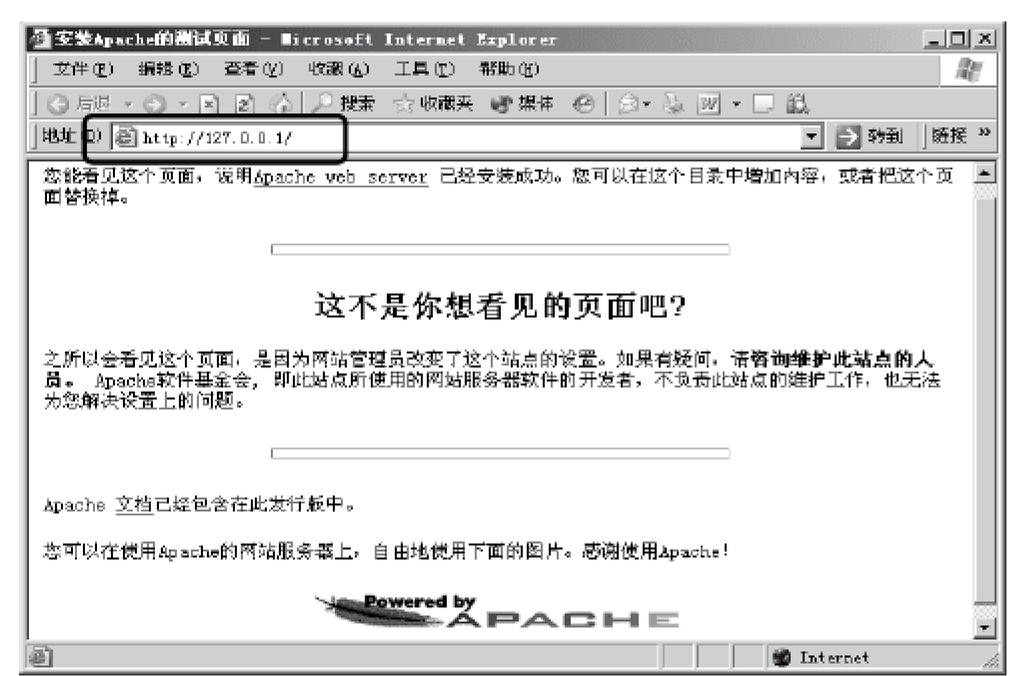


图 1.34　测试 Apache 服务器是否安装成功

（14）定位到 DocumentRoot 一行，可以将该路径修改为合适的路径，例如，设置为 DocumentRoot" C:\Apache\Apache2\webpage"，表示 Apache 服务器的虚拟目录为 C:\Apache\Apache2\webpage。

（15）将光标定位到 DirectoryIndex index.html index.html.var，在其后面添加一个 PHP 默认页，通常是 index.php，表示当访问该服务器时如果未指定要访问的 PHP 文件，则默认访问 index.php 文件。更改后的代码如下：

```
DirectoryIndex index.html index.html.var index.php
```

（16）为了使 Apache 识别 PHP 的扩展名，搜索并定位到 httpd.conf 文件的如下部分：

```
<Directory "C:/Apache/Apache2/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>
```

在后面添加如下两行代码：

```
AddType application/x-httpd-php .php .phtml .php3 .php4
AddType application/x-httpd-php-source .phps
```

（17）为了能够使用模块功能，模块通常以 DSO 方式构建，读者需要定位到如下代码：

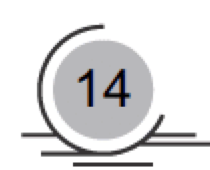
```
#LoadModule ssl_module modules/mod_ssl.so
```

使得能够在使用前获得指令的功能，然后以 module 方式加载 PHP，指向 PHP 5.0 目录下的 php5apache2.dll 文件，添加如下代码：

```
LoadModule php5_module c:\php5\php5apache2.dll
```

1.4.3　Windows 下 MySQL 的安装配置

PHP 能够支持市面上的大多数数据库，如 MySQL、Access、SQL Server、Oracle 等，而 MySQL 数据库是 PHP 程序开发人员公认的黄金搭档，这不仅因为 MySQL 是完全网络化的跨平台关系型数据库系统，而且它也是具有客户机/服务器体系结构的分布式数据库管理系统。它具有功能性强、使用简





洁、管理方便、运行速度快、版本升级快、安全性高等优点，而且 MySQL 数据库完全免费，从官方网站 http://www.mysql.com 即可免费下载到最新版本的 MySQL 安装包。

视频录像：TM\01\lx\Windows 下 MySQL 的安装配置.swf

在 Windows 操作系统下实现 MySQL 的安装配置分为以下几种方式：

☑ 如果下载的是 Windows Essentials（x86），例如 mysql-4.1.11-essential-win.exe，直接双击该文件安装即可，安装完毕后，在“开始”菜单中选择“运行”命令，在弹出的对话框中输入 cmd，进入“命令提示符”窗口，输入 mysqld-nt.exe，即可启动 MySQL 服务；如果还想详细地配置 MySQL，则需要进入 C:\mysql\bin 目录，运行 MySQLInstanceConfig.exe，并按提示操作即可。

☑ 如果下载的是 Windows（x86），例如 mysql-essential-5.0.24-win32.msi，解压该文件后双击 setup.exe 安装程序，即可逐步完成 MySQL 的安装。

☑ 如果下载的是 Without noinstall（unzip in C:\），直接将该文件解压到指定的安装目录下。在“开始”菜单中选择“运行”命令，在弹出的对话框中输入 cmd，然后进入“命令提示符”窗口，输入 mysqld-nt.exe 命令，即可启动 MySQL 服务器。

1.4.4　Windows 下 PHP 的安装配置

架设基于 PHP 的 Web 服务器，安装 PHP 是必需的。由于 PHP 的代码公开，所以其升级速度较快。安装 PHP 之前应从官方网站 http://www.php.net/ 下载最新版本的 PHP 安装程序，下面以 php-5.1.4-Win32.zip 为例讲解 PHP 的配置方法。

视频录像：TM\01\lx\Windows 下 PHP 的安装配置.swf

在 Windows 操作系统下实现 PHP 的安装配置的操作步骤如下：

（1）成功下载到 PHP 的安装包后，首先应对其进行解压，如解压到 C:\php5 目录下，如图 1.35 所示。

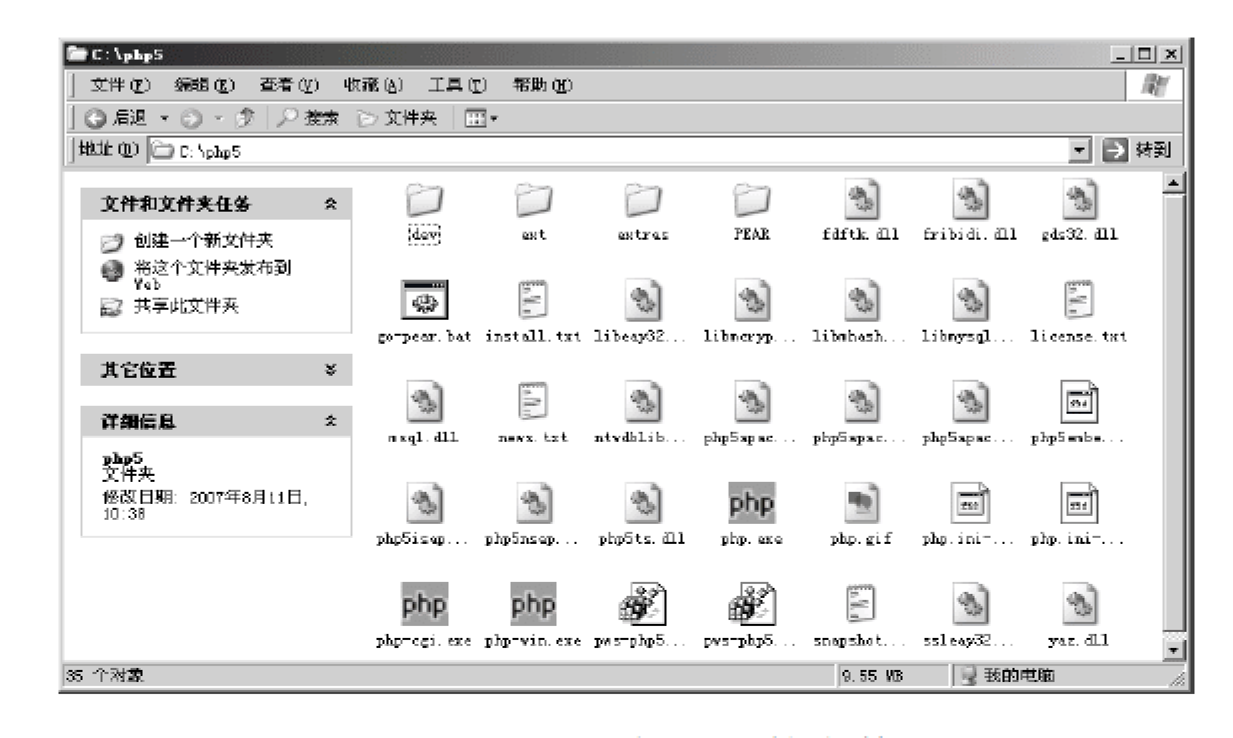


图 1.35　PHP 解压后的文件

（2）将 C:\php5\目录下的所有.dll 文件复制到 C:\Windows\system32\（如果是 Windows 2000 操作系统，则为 C:\WINNT\system32\）目录下。

（3）将 C:\php5\目录下的 php.ini-dist 文件复制到 C:\Windows\（如果是 Windows 2000 操作系统，则为 C:\WINNT\）目录下，然后将 php.ini-dist 重命名为 php.ini，并用记事本打开该文件进行编辑，具体编辑方式如下：

首先，定位到该文件 register_globals = Off 所在行，将 Off 改为 On。代码如下：

```
register_globals = On
```

其次，定位到该文件 extension_dir=".\"所在行，将路径改为"C:\php5\ext"。代码如下：

```
extension_dir="C:\php5\ext"
```

最后，查找定位 Windows Extensions，将下面两行代码：

```
; extension=php_gd2.dll
; extension=php_mysql.dll
```

前面的注释（即分号标识符）去掉，这样 PHP 就可以支持 GD2 函数库和 MySQL 相关函数，如果想让 PHP 支持其他功能函数，同样需要将与这些操作函数相关的 dll 文件前的注释去掉。

注意 以上为 PHP 的常规配置，在 PHP 的官方论坛上可以查到更多的详细配置。由于 PHP 的版本有多种，所以 php.ini 的配置也存在差别，具体使用时应根据实际情况配置 php.ini 文件。

（4）PHP 配置完成以后，需要重新启动 Apache 服务器，然后编写 test.php 文件并输入如下代码：

```
<?php
  phpinfo();
?>
```

将文件保存到 Apache 安装目录下的 webpage 子目录下，然后在浏览器中输入 http://127.0.0.1/test.php，如果显示如图 1.36 所示的页面，则说明 PHP 配置成功。

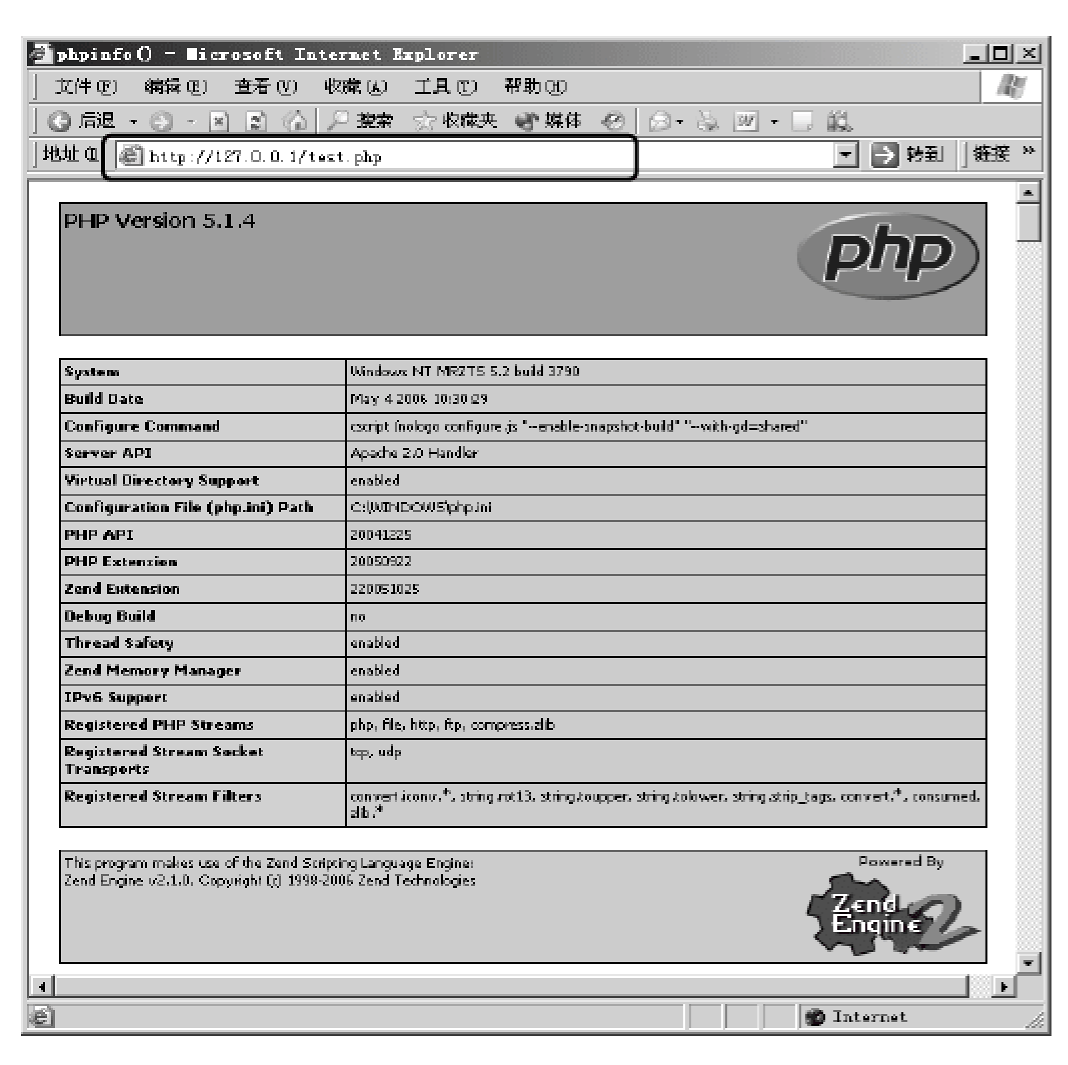
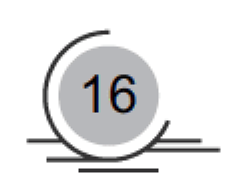


图 1.36　测试成功

技巧 安装文件的路径也要遵循一定的客观原则，为了避免在 Windows 和 Linux 间移植程序时带来不便，选择 D:\usr\local\php 的目录时要和在 Linux 下的安装目录相匹配。建议最好不要选择中间有空格的目录，如 E:\Program Files\PHP，这样做会导致一些未知错误或崩溃发生。

1.4.5　在 Windows 下设置 IIS+PHP 的执行环境

PHP 可以在很多服务器下运行，下面介绍 IIS 的安装方法以及 PHP 是如何在 IIS 服务器下运行的。





1．IIS 的安装

Internet 信息服务器（Internet Information Server，缩写为 IIS）是 Microsoft 的 Web 服务器，它集成于 Windows NT Server 之中，方便易用，为 Web 应用程序提供了功能强大的运行平台。

IIS 目前最新版本为 6.0，在 Windows 2003 Server 操作系统中默认安装，如果没有安装 Windows 2003 Server，可以选择 IIS 5.0，它可以在 Windows XP 或者 Windows 2000 操作系统中安装，对于开发而言，IIS 的各个版本之间的区别并不大。

（1）在 Windows XP 或者 Windows 2000 操作系统下安装 IIS

视频录像：TM\01\lx\在 Windows XP 或者 Windows 2000 操作系统下安装 IIS.swf

选择“开始”/“设置”/“控制面板”命令，启动“添加/删除程序”应用程序，单击“添加/删除 Windows 组件”图标，在弹出的“Windows 组件向导”对话框的“组件”列表框中选择“Internet 信息服务（IIS）”选项，最后单击“下一步”按钮，即可完成 IIS 的安装。

（2）在 Windows 2003 Server 服务器下安装 IIS

视频录像：TM\01\lx\在 Windows 2003 Server 服务器下安装 IIS.swf

首先选择“开始”/“控制面板”命令，然后启动“添加/删除程序”应用程序，单击“添加/删除 Windows 组件”图标，在弹出的“Windows 组件向导”对话框的“组件”列表框中选择“应用程序服务器”选项，然后单击“详细信息”按钮，选择“Internet 信息服务（IIS）”选项，最后单击“下一步”按钮，即可完成 IIS 的安装。

注意　若想在 IIS 中支持 PHP 的运行环境，在安装 IIS 时，需在“网站访问权限”窗口中选中读取、运行脚本（如 ASP）、执行（如 ISAPI 应用程序或 CGI）选项。

2．架设 PHP 到 IIS 服务器

在架设 PHP 到 IIS 服务器之前，首先应确保 PHP 已经正确安装到系统中。本节将以 Windows 2003 Server 服务器中的 IIS 6.0 为例，介绍架设 PHP 到 IIS 服务器的方法。

视频录像：TM\01\lx\在 Windows 下设置 IIS+PHP 的执行环境.swf

（1）选择“开始”/“管理工具”/“Internet 信息服务（IIS）管理器”命令，打开如图 1.37 所示的 Internet 信息服务窗口。

（2）右击“默认网站”节点，在弹出的快捷菜单中选择“属性”命令，将打开如图 1.38 所示的对话框。

（3）单击图 1.38 中“主目录”选项卡中的“配置”按钮，打开如图 1.39 所示的对话框。

（4）单击图 1.39 中的“添加”按钮，将打开如图 1.40 所示的对话框，单击“可执行文件”文本框后的“浏览”按钮，在打开的窗口中选择 php5 目录下的 php5isapi.dll 文件，在“扩展名”文本框中输入“.php”，然后单击“确定”按钮，实现扩展名的映射。

（5）单击“主目录”选项卡中的“浏览”按钮，选择网站的路径，如图 1.41 所示。

（6）在“Internet 信息服务（IIS）管理器”窗口中单击“Web 服务扩展”节点，在“Web 服务扩展”区域中选择“所有未知 ISAPI 扩展”选项，然后单击“允许”按钮，将该服务的状况由“禁止”设置为“允许”，如图 1.42 所示。

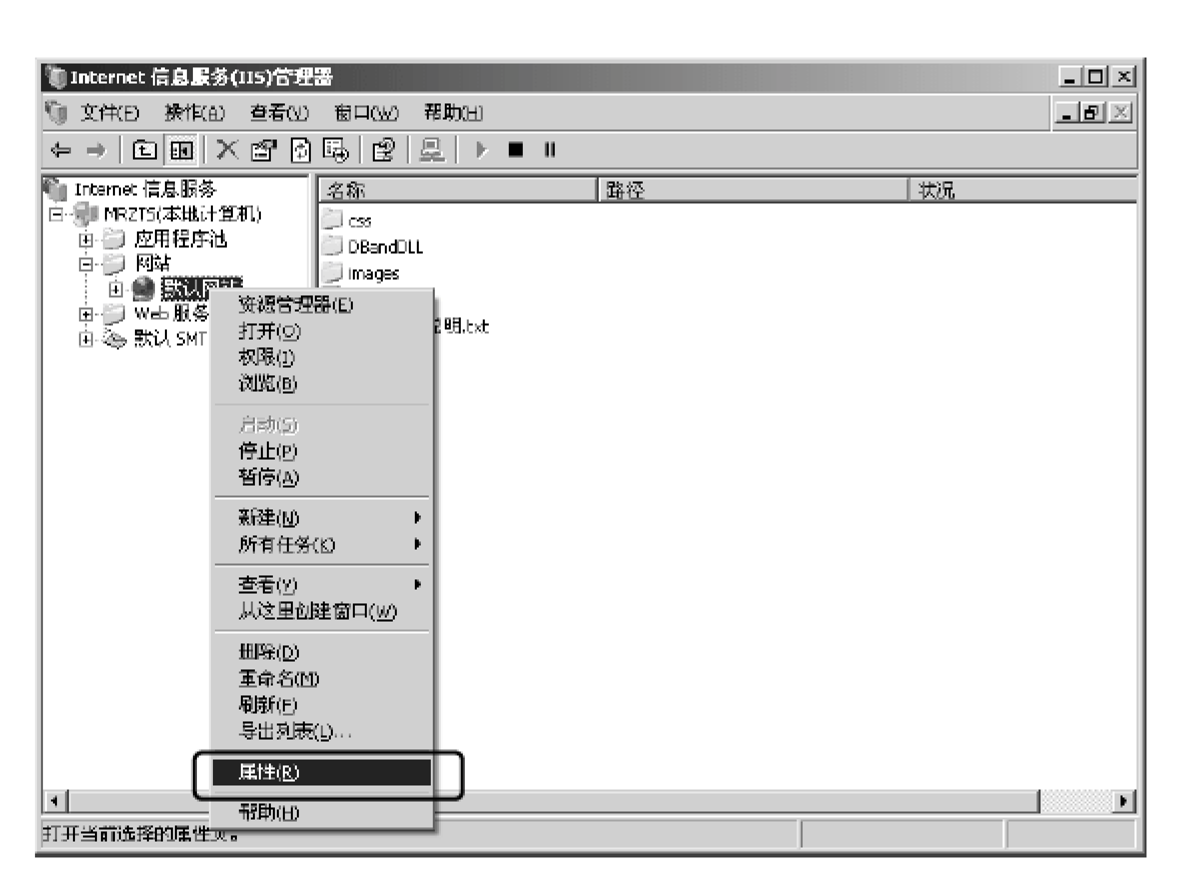


图 1.37　Internet 信息服务窗口

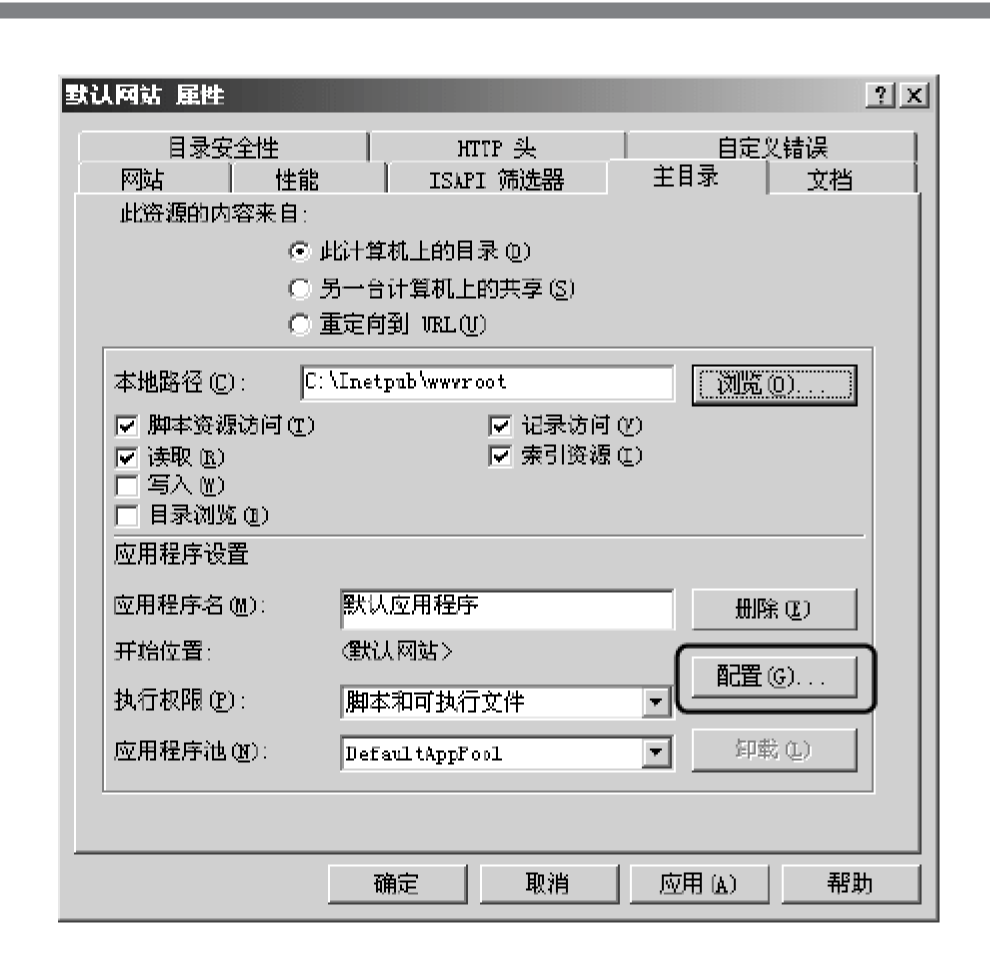


图 1.38　默认 Web 站点属性



图 1.39　应用程序配置



图 1.40　添加/编辑应用程序扩展名映射

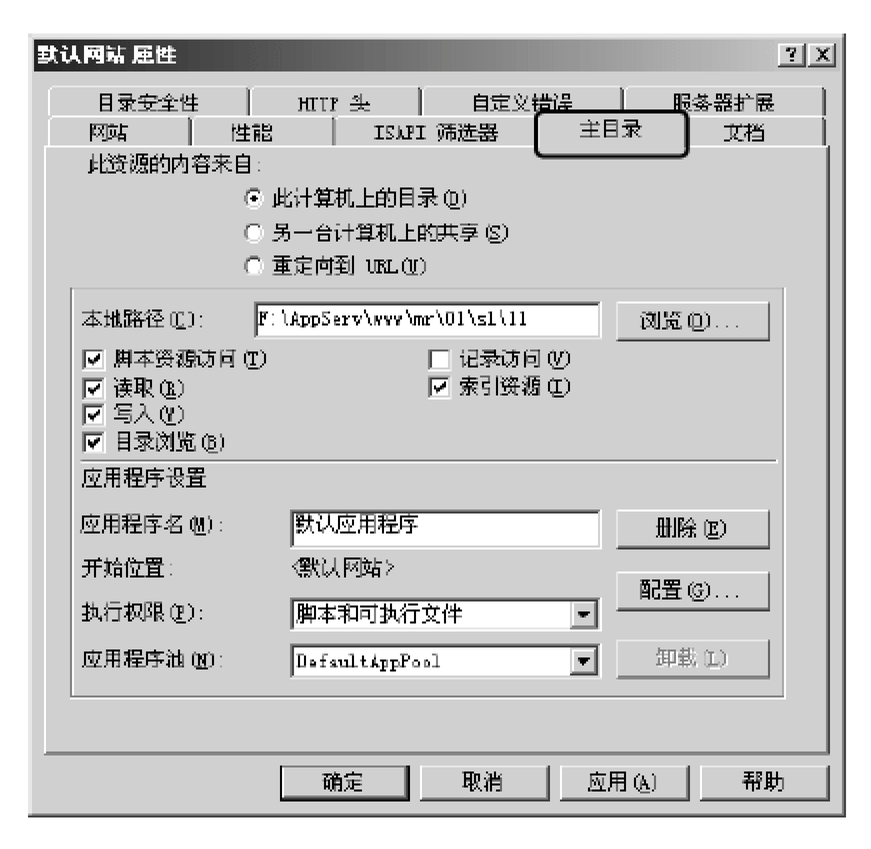


图 1.41　设置网站的路径



图 1.42　设置 Web 服务扩展

（7）打开 IIS 服务器，浏览 index.php 文件，如果在浏览器中打开网页文件，则说明 IIS+PHP 运行环境配置成功，如图 1.43 所示。

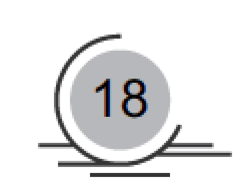




图 1.43　IIS+PHP 运行环境配置成功

1.5　数据库设计

1.5.1　数据库分析

本系统是一个中小型的供求信息平台，但是由于平台会涉及海量数据，因此需要充分考虑到成本问题及用于需求（如跨平台）等问题。而 MySQL 是世界上最为流行的开放源码的数据库，是完全网络化的、跨平台的关系型数据库系统，这正好满足了中小型企业的需求，所以本系统采用 MySQL 数据库。

1.5.2　数据库概念设计

根据前面对系统所做的需求分析、系统设计，规划出本系统中使用的数据库实体分别为免费信息实体、付费信息实体、广告信息实体和管理员实体。下面分别介绍这几个实体的 E-R 图。

1. 免费信息实体

免费信息实体包括编号、信息类型、信息标题、信息内容、联系人、联系电话、审核状态和发布时间属性。其中审核状态属性用来标识信息是否审核，“1”表示“是”，“0”表示“否”。免费信息实体的 E-R 图如图 1.44 所示。

2. 付费信息实体

付费信息实体包括编号、信息类型、信息标题、信息内容、联系人、联系电话、发布时间、截止时间和审核状态属性。其中审核状态属性用来标识信息是否付费，“1”表示“是”，“0”表示“否”。付费信息实体的 E-R 图如图 1.45 所示。

3. 广告信息实体

广告信息实体包括编号、信息标题、信息内容、发布时间和推荐状态属性。其中推荐状态属性用来标识信息是否在前台显示，“1”表示“是”，“0”表示“否”。广告信息实体的 E-R 图如图 1.46 所示。

4. 管理员实体

管理员实体包括编号、管理员名和加密密码属性。管理员实体的 E-R 图如图 1.47 所示。

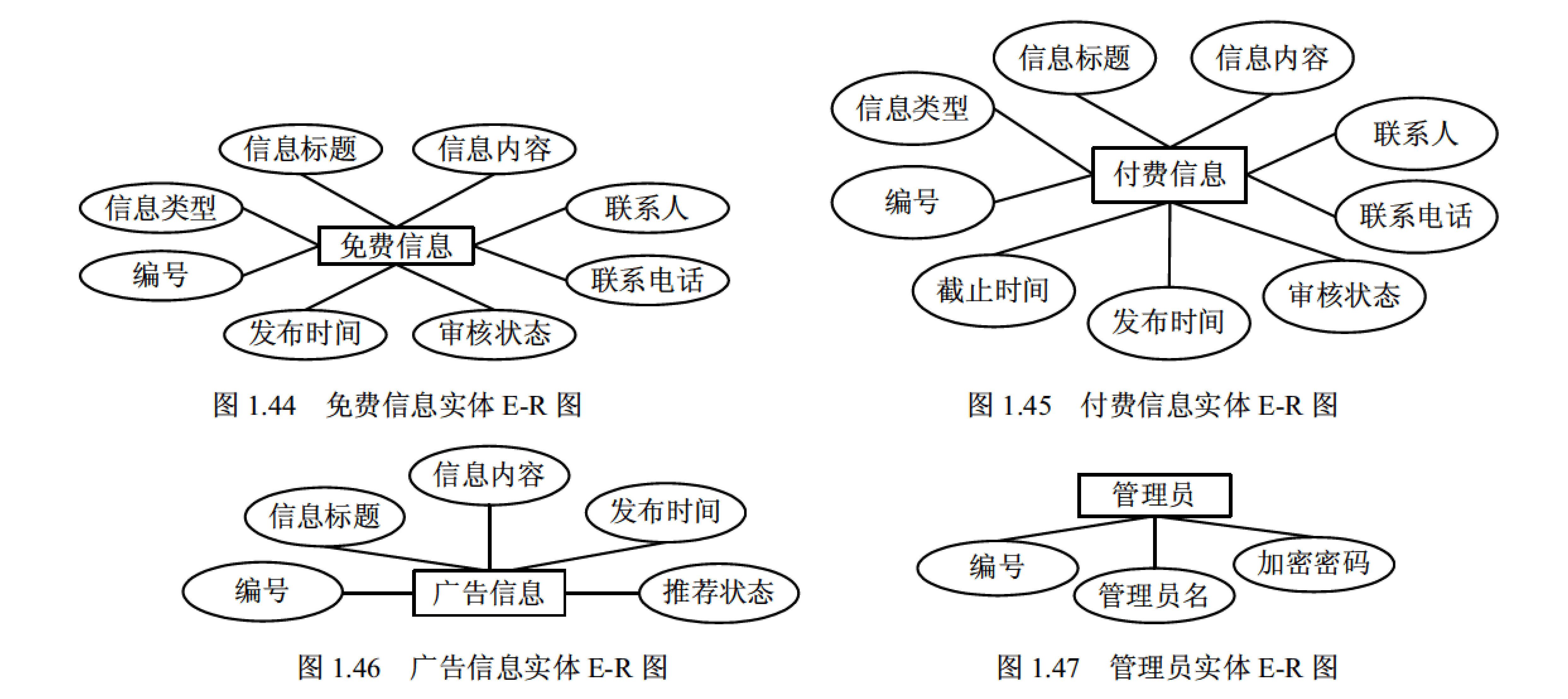


图 1.44　免费信息实体 E-R 图

图 1.45　付费信息实体 E-R 图

图 1.46　广告信息实体 E-R 图

图 1.47　管理员实体 E-R 图

1.5.3　使用 PowerDesigner 建模

在数据库概念设计中已经分析了本系统中主要的数据实体对象，通过这些实体可以得出数据表结构的基本模型，最终实施到数据库中，形成完整的数据结构。下面将介绍使用 PowerDesigner 工具完成本系统的数据库建模。

（1）运行 PowerDesigner，并在 PowerDesigner 主窗口中选择 File/New 命令，在打开的 New 对话框中选择 Physical Data Model（物理数据模型，PDM）列表项，单击 OK 按钮，打开如图 1.48 所示的 Choose DBMS（选择数据库管理系统）对话框。

（2）在 DBMS 下拉列表框中选择 MySQL 数据库，这里选择 MySQL 3.23 版本，这是由于当前使用的 PowerDesigner 的版本还不支持 MySQL 5.0，但是随着 PowerDesigner 版本的升级将支持更多的 DBMS。

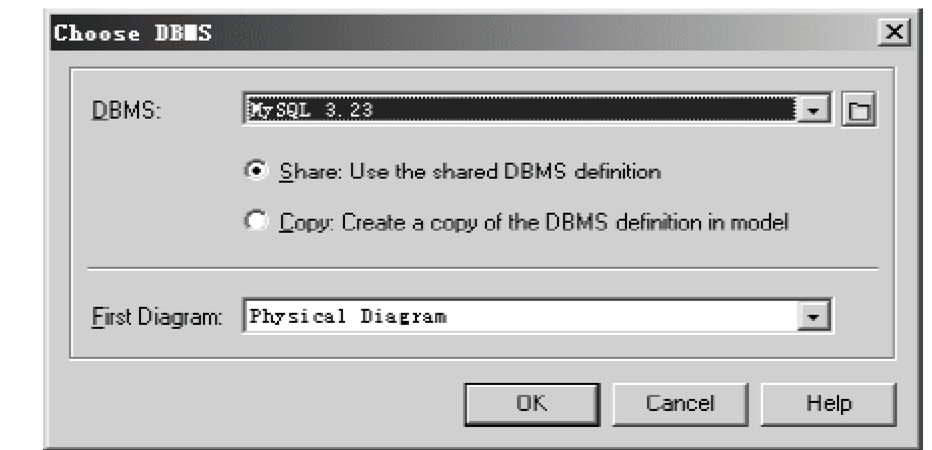


图 1.48　Choose DBMS 对话框

（3）其他选项采用默认设置即可，单击 OK 按钮，打开新建的 PDM 窗口。在该窗口的上方为空的图形窗口，下方为输出窗口。其中图形窗口的右侧有一个工具面板，如图 1.49 所示。

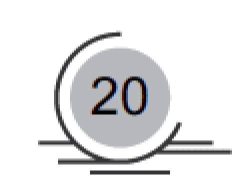
（4）在图 1.49 中单击“建立表”图标，这时鼠标指针将显示为▭，在图形窗口的合适位置单击鼠标左键，此时在图形窗口中将显示如图 1.50 所示的表符号。



图 1.49　工具面板

Table_1

图 1.50　表符号





注意　细心的读者可以发现，此时的鼠标指针仍然是，如果再单击鼠标左键还将出现类似图 1.50 所示的表符号，如果想取消该指针，可以单击工具面板中的指针图标。

（5）在图 1.50 所示的表符号上双击鼠标左键，将打开 Table Properties（表属性）对话框，默认情况下选择的是 General 选项卡，在该选项卡的 Name 文本框中输入表的名字 tb_admin，此时在 Code 文本框中也将自动显示 tb_admin，其他选项选择默认即可。

（6）选择 Columns 选项卡，首先单击列输入列表的第一行，将自动填写一行信息，然后将 Name 列修改为 id，同时 Code 列也将自动显示为 id，再在 Data Type 列中选择 integer unsigned 列表项，最后选中 P 列的复选框，此时 M 列的复选框也将自动选中。

（7）按照步骤（6）的方法再添加两个列 name 和 pwd，如图 1.51 所示。

（8）在图 1.51 中单击“应用”按钮后，双击 id 列的内容，将打开 Column Properties（列属性）对话框，如图 1.52 所示。默认选择 General 选项卡，在该选项卡中选中 Identity 复选框，此项操作用于设置 id 列为自动编号列。

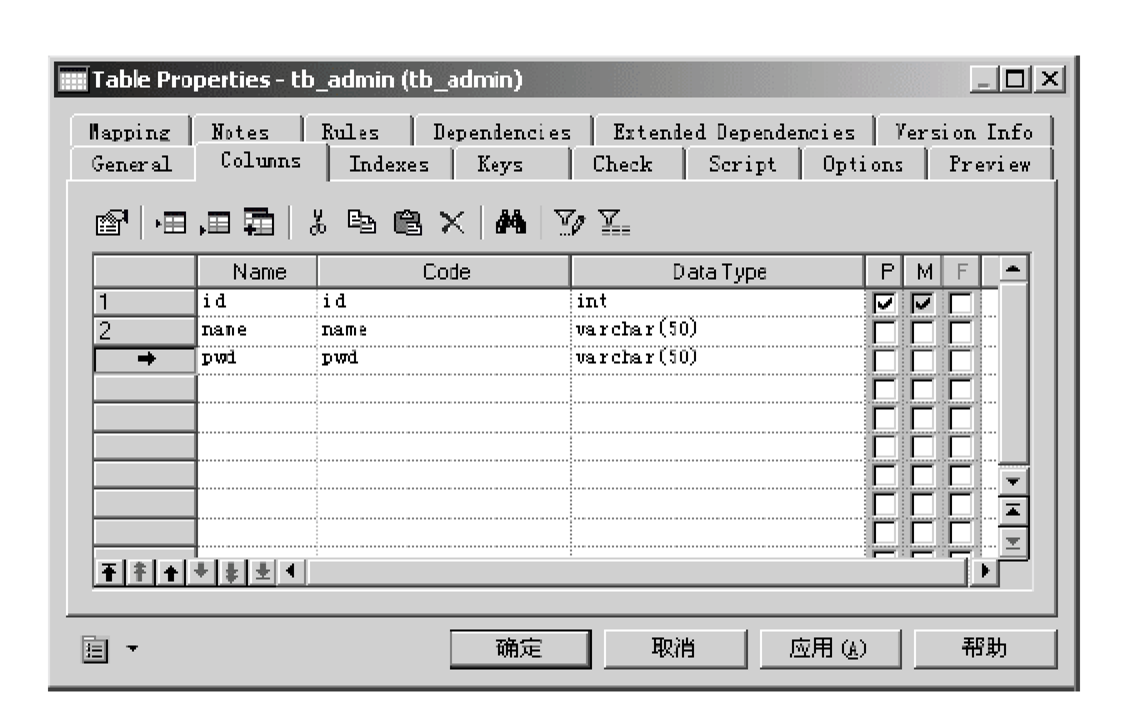


图 1.51　Columns 选项卡



图 1.52　Column Properties 对话框

（9）单击“应用”按钮后，再单击“确定”按钮，关闭 Column Properties 对话框。

（10）单击“确定”按钮，关闭 Table Properties 对话框，完成 tb_admin 表的创建。

（11）按照步骤（4）～（10）的方法创建本系统中的其他数据表。创建完成的模型图如图 1.53 所示。

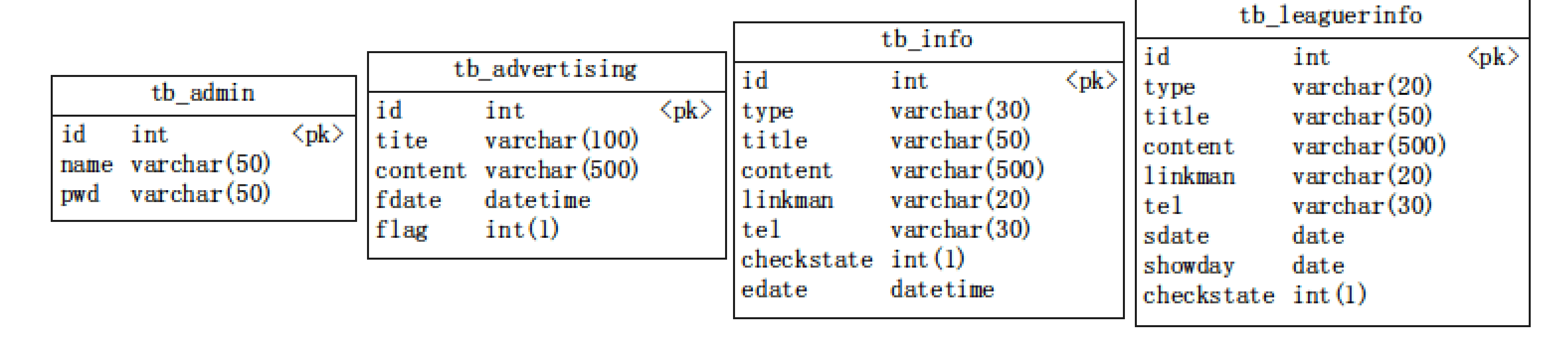


图 1.53　九九度供求信息网的模型图

技巧　在默认情况下，创建后的表符号中的全部文字均为常规样式的宋体 8 号字，如果想修改文字的格式，可以选中全部表符号，按 Ctrl+T 键，在打开的 Symbol Format 对话框中选择 Font 选项卡，在该选项卡中设置相关内容的字体及样式和字号等。

（12）选择 PowerDesigner/Database/Generate Database 命令，将打开 Database Generation 对话框，在该对话框中设置导出的脚本文件的名称（如 pursey.sql）及保存路径（如 F:\ pursey），在下方的各个选项卡中设置相关的导出内容，设置完毕后，单击“确定”按钮，导出数据库脚本文件完成。

说明 通过以上方法导出的脚本文件，可以在 MySQL 的客户端命令行窗口中通过 source 命令执行，并创建相应的数据表。

1.5.4 创建数据库及数据表

结合实际情况及对用户需求的分析，可知九九度供求信息网中应用的 db_pursey 数据库主要包含如下 4 个数据表，如图 1.54 所示。



服务器: localhost ▸ 数据库: db_pursey

表	类型	整理	说明
tb_admin	MyISAM	gb2312_chinese_ci	管理员信息表
tb_advertising	MyISAM	gb2312_chinese_ci	企业广告信息表
tb_info	MyISAM	gb2312_chinese_ci	免费供求信息表
tb_leaguerinfo	MyISAM	gb2312_chinese_ci	付费供求信息表

图 1.54　九九度供求信息网数据表

各数据表的表结构如图 1.55～图 1.58 所示。

1．tb_admin（管理员信息表）

管理员信息表主要用于存储管理员的信息。该数据表的结构如图 1.55 所示。

2．tb_advertising（企业广告信息表）

企业广告信息表主要用于存储企业发布的广告信息。该数据表的结构如图 1.56 所示。

服务器: localhost ▸ 数据库: db_pursey ▸ 表 : tb_admin

字段	类型	整理	Null	默认	额外	说明
id	int(4)		否		auto_increment	自动编号
name	varchar(50)	gb2312_chinese_ci	是	NULL		管理员名
pwd	varchar(50)	gb2312_chinese_ci	是	NULL		管理员密码

图 1.55　管理员信息表结构

服务器: localhost ▸ 数据库: db_pursey ▸ 表 : tb_advertising

字段	类型	整理	Null	默认	额外	说明
id	int(4)		否		auto_increment	自动编号
title	varchar(100)	gb2312_chinese_ci	否			广告标题
content	varchar(500)	gb2312_chinese_ci	否			广告内容
fdate	datetime		否			发布时间
flag	int(1)		否	0		推荐状态

图 1.56　企业广告信息表结构

3．tb_info（免费供求信息表）

免费供求信息表主要用于存储用户免费发布的供求信息。该数据表的结构如图 1.57 所示。

4．tb_leaguerinfo（付费供求信息表）

付费供求信息表主要用于存储付费的供求信息。该数据表的结构如图 1.58 所示。

服务器: localhost ▸ 数据库: db_pursey ▸ 表 : tb_info

字段	类型	整理	Null	默认	额外	说明
id	int(4)		否		auto_increment	自动编号
type	varchar(30)	gb2312_chinese_ci	否			信息类型
title	varchar(50)	gb2312_chinese_ci	否			信息标题
content	varchar(500)	gb2312_chinese_ci	否			信息内容
linkman	varchar(20)	gb2312_chinese_ci	否			联系人
tel	varchar(30)	gb2312_chinese_ci	否			联系电话
checkstate	int(1)		否	0		审核状态
edate	datetime		否			发布时间

图 1.57　免费供求信息表结构



服务器: localhost ▸ 数据库: db_pursey ▸ 表 : tb_leaguerinfo

字段	类型	整理	Null	默认	额外	说明
id	int(4)		否		auto_increment	自动编号
type	varchar(20)	gb2312_chinese_ci	否			信息类型
title	varchar(50)	gb2312_chinese_ci	否			信息标题
content	varchar(500)	gb2312_chinese_ci	否			信息内容
linkman	varchar(20)	gb2312_chinese_ci	否			联系人
tel	varchar(30)	gb2312_chinese_ci	否			联系电话
sdate	date		否			发布日期
showday	date		否			截止日期
checkstate	int(1)		否	0		审核状态

图 1.58　付费供求信息表结构





1.6　单元测试

在现代软件开发过程中，测试不再作为一个独立的生命周期，而是成为与编写代码同步进行的开发活动。单元测试能够提高程序员对程序的信心，保证程序的质量，加快软件开发速度，使程序易于维护。

1.6.1　单元测试概述

在程序设计过程中会有多种测试，单元测试只是其中的一种。单元测试并不能保证程序是完美无缺的，但是在所有的测试中，单元测试是第一个环节，也是最重要的一个环节。单元测试是一种由程序员自行测试的工作。简单地说，单元测试就是测试代码撰写者依据其所设想的方式执行是否产生了预期的结果。

单元测试不仅是无错编码的一种辅助手段，还必须是可重复的，即无论是在软件修改或是移植到新的运行环境的过程中都可以进行单元测试。

与单元测试有密切关联的开发活动包括代码走读（Code review）、静态分析（Static analysis）和动态分析（Dynamic analysis）。代码走读就是对软件的源代码进行宏观阅读，整理开发思路。静态分析就是对软件的源代码进行研读，查找错误或收集一些度量数据，并不需要对代码进行编译和执行。动态分析就是通过观察软件运行时的动作，来提供执行跟踪、时间分析以及测试覆盖度方面的信息。

1.6.2　单元测试的优点

单元测试具有以下优点：

- ☑ 一种验证行为。程序中的每一项功能都是用测试来验证它的正确性，为以后的开发提供支援。就算是开发后期，也可以轻松地增加功能或更改程序结构，而不用担心这个过程中会破坏重要的东西，而且它为代码的重构提供了保障。这样，我们就可以更自由地对程序进行改进。
- ☑ 一种设计行为。编写单元测试将使用户从调用者观察、思考。特别是先写测试（test-first），迫使设计者把程序设计成易于调用和可测试的，即解除软件中的耦合。
- ☑ 一种编写文档的行为。单元测试是一种无价的文档，它是展示函数或类如何使用的最佳文档。这份文档是可编译、可运行的，并且它保持最新，永远与代码同步。
- ☑ 具有回归性。自动化的单元测试避免了代码出现回归，编写完成之后，可以随时随地地快速运行测试。

1.7　前台首页设计

1.7.1　前台首页概述

网站主页是关于网站的建设及形象宣传，它对网站生存和发展起着非常重要的作用，应该是一个信息含量较高、内容较丰富的宣传平台。九九度供求信息网前台首页主要包含以下内容：

☑ 网站菜单导航（包括公寓信息、招聘信息、求职信息、培训信息、家教信息、房屋信息、车辆信息、求购信息、出售信息、招商引资、寻人/物启示等）。

☑ 发布免费的供求信息（包括公寓信息、招聘信息、求职信息、培训信息、家教信息、房屋信息、车辆信息、求购信息、出售信息、招商引资、寻人/物启示等）。

☑ 推荐供求信息显示（包括公寓信息、招聘信息、求职信息、培训信息、家教信息、房屋信息、车辆信息、求购信息、出售信息、招商引资、寻人/物启示等），其中，付费信息按时间顺序降序排列，免费信息按时间顺序分页显示。

☑ 显示推荐的企业广告信息。

☑ 供求信息快速检索，支持模糊查询和查询关键字描红功能。

☑ 后台登录入口，为管理员进入后台提供一个入口。

下面看一下本案例中提供的前台首页，该页面在本书光盘中的路径为\TM\01\99pursey\index.php，如图 1.59 所示。



图 1.59　九九度供求信息网首页

各区域的介绍及所对应的 PHP 文件如表 1.3 所示。

表 1.3　页面框架中各区域介绍及对应的 PHP 文件

名　称	说　明	对应的 PHP 文件
导航栏	提供查看各类信息的超链接	top.php
信息检索区	企业广告显示及各类信息检索	left.php
内容显示区	根据用户请求显示相应内容	根据请求加载相应的 PHP 文件，默认加载 main.php
版权区	显示版权信息	bottom.php





1.7.2　前台首页技术分析

超链接在本质上属于一个网页的一部分，它是一种允许用户同其他网页或站点之间进行连接的元素。各个网页链接在一起后，才能真正构成一个网站。按照使用对象的不同，网页中的链接可以分为文本超链接、图像超链接、E-mail 链接、锚点链接、多媒体文件链接和空链接等。

图像不但可以建立超链接，还可以实现图像映射。图像映射是指一幅图像可以建立多个超链接，即在图像上定义多个区域，每个区域链接到不同的地址，这样的区域称为热区（也称为热点，Hotspot）。本项目就是应用图像热区实现的功能导航。

图像映射有两种：服务器端映射（Server-side Image Map）和客户端映射（Client-side Image Map）。目前使用最多的是客户端映射，因为客户端映射使图像上对应区域的坐标以及链接的 URL 地址都在浏览器端读入，省去和服务器之间互传坐标和 URL 的时间。

在 PHP 文件中，使用<MAP>标记创建图像映射。语法如下：

```
<IMG SRC="file_name" USEMAP="#MapName">
<MAP NAME="MapName">
    <AREA SHAPE="value" COORDS="坐标" HREF="URL" ALT="描述文字">
    <AREA SHAPE="value" COORDS="坐标" HREF="URL" ALT="描述文字">
    …
</MAP>
```

在<IMG>标记中设置属性 USEMAP，确定创建图像映射。

<MAP>标记的属性如表 1.4 所示。

表 1.4　<MAP>标记的属性

<MAP>标记的属性	描　述
NAME	图像映射的名称
SHAPE	定义图像映射区域的名称
COORDS	设定区域坐标
HREF	设定区域的链接地址
ALT	设定区域链接的描述文字

在<MAP>标记中，根据属性 SHAPE 的取值不同，相应坐标的设定也不同。下面介绍属性 SHAPE 的 3 种取值以及相应坐标的设定。

- ☑ 设定属性 SHAPE 的属性值为 rect。属性 SHAPE 取值为 rect，表示矩形区域，属性 COORDS 的坐标形式为“X1，Y1，X2，Y2”。其中，X1、Y1 代表矩形左上角的 X 坐标和 Y 坐标，X2、Y2 代表矩形右下角的 X 坐标和 Y 坐标。
- ☑ 设定属性 SHAPE 的属性值为 circle。属性 SHAPE 取值为 circle，表示圆形区域，属性 COORDS 的坐标形式为“X，Y，r”。其中，X、Y 为圆心坐标，r 为圆的半径。
- ☑ 设定属性 SHAPE 的属性值为 poly。属性 SHAPE 取值为 poly，表示多边形区域，属性 COORDS 的坐标形式为“X1，Y1，X2，Y2，…”。其中，Xn、Yn 代表构成多边形每一点的坐标值，n 的取值为 1，2，3，…多边形有几边就有几对 X、Y 坐标。

下面在 Dreamweaver 编辑器中创建图像映射，单击图像不同区域，链接到不同的文件。创建图像

映射的步骤如下：

（1）在 Dreamweaver 编辑器中选中所要编辑的图像，在属性窗口中，有 3 种形状的热点工具，分别为矩形热点工具、圆形热点工具、多边形热点工具。

（2）选择一种热点工具，在图像上按住鼠标左键拖动，确定所选区域。

（3）设置区域属性，在“链接”文本框中输入所要链接书签的名称，在“替代”文本框中输入区域链接文件，如 index.php。

（4）重复步骤（2）和步骤（3），在图像上定义不同的区域链接。

1.7.3 前台首页的实现过程

本系统中所有的前台页面都采用了二分栏结构，分为导航栏、信息检索区、内容显示区和版权区 4 个区域。为了方便网站的日后维护，将这 4 个区域形成单独的 PHP 文件，然后应用 include 语句将这 4 个文件包含进来。前台首页文件的代码如下：

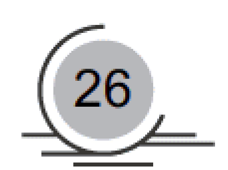
例程01 代码位置：光盘\TM\01\99pursey\index.php

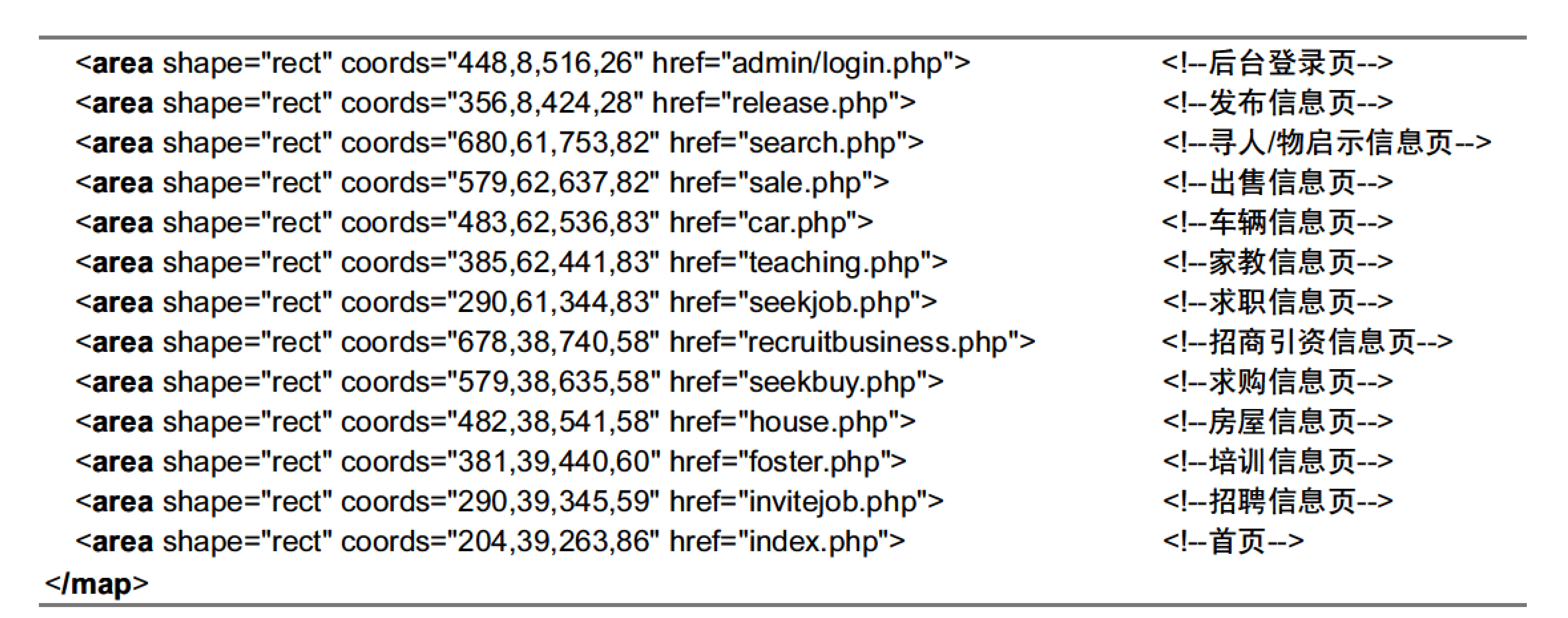
```
<table width="780" border="0" align="center" cellpadding="0" cellspacing="0">
  <tr valign="top">
    <td colspan="2"><?php include("top.php");?></td>                    <!--包含导航文件-->
  </tr>
  <tr>
     <!--包含信息检索文件-->
     <td width="217" valign="top" background="Images/line2.gif"><?php include("left.php");?></td>
     <!--包含内容显示文件-->
    <td width="586" valign="top" bgcolor="#FEFEF6"><?php include("main.php");?></td>
  </tr>
  <tr>
    <td colspan="2"><?php include("bottom.php");?></td>                 <!--包含版权信息文件-->
  </tr>
</table>
```

其中，导航文件 top.php 页应用了图像映射来创建文件的超链接，代码如下：

例程02 代码位置：光盘\TM\01\99pursey\top.php

```
<table width="780" height="201" border="0" align="center" cellpadding="0" cellspacing="0">
  <tr>
    <td height="38"><img src="Images/banner.gif" width="780" height="202" border="0" usemap="#Map"></td>
  </tr>
</table>
<!-- ---------------------------------------------图像映射---------------------------------- -->
<map name="Map">
     <!-- ----------------------------------------联系我们---------------------------------- -->
       <area shape="rect" coords="685,8,744,27" href="mailto:99pursey@pursey**.com">
     <!-- ----------------------------------------设为首页---------------------------------- -->
  <area shape="rect" coords="613,9,666,27"
onClick="this.style.behavior='url(#default#homepage)';this.setHomePage('http://localhost/99pursey/index.php');">
     <!-- ----------------------------------------加入收藏----------------------------------- -->
       <area shape="rect" coords="535,8,593,26"
href="javascript:window.external.AddFavorite('http://localhost/99pursey/index.php','九九度供求信息网')">
```





```
    <area shape="rect" coords="448,8,516,26" href="admin/login.php">               <!--后台登录页-->
    <area shape="rect" coords="356,8,424,28" href="release.php">                   <!--发布信息页-->
    <area shape="rect" coords="680,61,753,82" href="search.php">                   <!--寻人/物启示信息页-->
    <area shape="rect" coords="579,62,637,82" href="sale.php">                     <!--出售信息页-->
    <area shape="rect" coords="483,62,536,83" href="car.php">                      <!--车辆信息页-->
    <area shape="rect" coords="385,62,441,83" href="teaching.php">                 <!--家教信息页-->
    <area shape="rect" coords="290,61,344,83" href="seekjob.php">                  <!--求职信息页-->
    <area shape="rect" coords="678,38,740,58" href="recruitbusiness.php">          <!--招商引资信息页-->
    <area shape="rect" coords="579,38,635,58" href="seekbuy.php">                  <!--求购信息页-->
    <area shape="rect" coords="482,38,541,58" href="house.php">                    <!--房屋信息页-->
    <area shape="rect" coords="381,39,440,60" href="foster.php">                   <!--培训信息页-->
    <area shape="rect" coords="290,39,345,59" href="invitejob.php">                <!--招聘信息页-->
    <area shape="rect" coords="204,39,263,86" href="index.php">                    <!--首页-->
</map>
```

1.8　免费供求信息发布模块设计

1.8.1　免费供求信息发布模块概述

免费供求信息的发布提供对象为供求信息用户，是供求信息网站非常重要的功能，也是供求信息网站的核心功能。

免费供求信息发布模块可以完成 11 种不同类别信息的发布。用户可以根据自身需要将供求信息发布到相应的信息类别中（共包括 11 个信息类别：公寓信息、招聘信息、求职信息、培训信息、家教信息、房屋信息、车辆信息、求购信息、出售信息、招商引资、寻人/物启示等类型供求信息）。信息发布成功后，需要管理员进行审核，只有审核成功的信息才能显示在前台相应的信息类别网页中。免费供求信息发布的流程如图 1.60 所示。

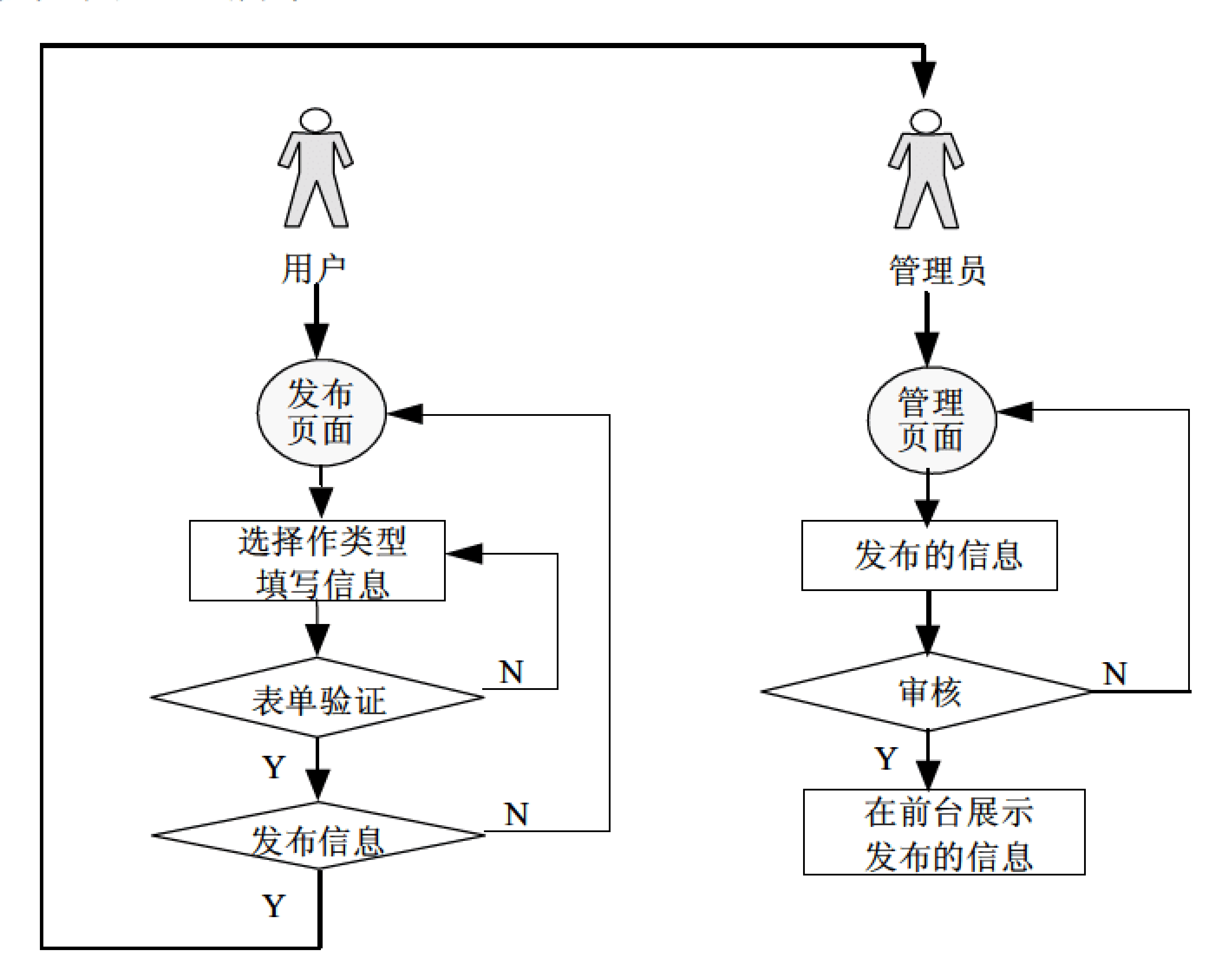


图 1.60　免费供求信息发布流程图

1.8.2 免费供求信息发布模块技术分析

本模块实现免费供求信息发布功能，主要应用到如下几个函数。

1. mysql_connect()函数

打开一个到 MySQL 服务器的连接。如果成功则返回一个 MySQL 连接标识，失败则返回 false。语法如下：

```
resource mysql_connect ( [string server [, string username [, string password [, bool new_link [, int client_flags]]]]] )
```

mysql_connect()函数的参数说明如表 1.5 所示。

表 1.5 mysql_connect()函数的参数说明

参　数	说　明
server	MySQL 服务器。可以包括端口号，例如"hostname:port"，或者到本地套接字的路径，例如对于 localhost 的":/path/to/socket"。如果 PHP 指令 mysql.default_host 未定义（默认情况），则默认值是'localhost:3306'
username	用户名。默认值是服务器进程所有者的用户名
password	密码。默认值是空密码
new_link	如果用同样的参数第二次调用 mysql_connect()函数，将不会建立新连接，而是返回已经打开的连接标识。参数 new_link 改变此行为并使 mysql_connect()函数总是打开新的连接，甚至当 mysql_connect()函数曾在前面被同样的参数调用过
client_flags	client_flags 参数可以是以下常量的组合：MYSQL_CLIENT_SSL、MYSQL_CLIENT_COMPRESS、MYSQL_CLIENT_IGNORE_SPACE 或 MYSQL_CLIENT_INTERACTIVE

2. mysql_select_db()函数

选择 MySQL 数据库。如果成功返回 true，失败返回 false。语法如下：

```
bool mysql_select_db ( string database_name [, resource link_identifier] )
```

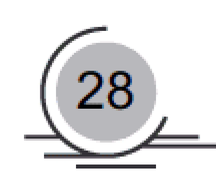
mysql_select_db()函数设定与指定的连接标识符所关联的服务器上的当前激活数据库。如果没有指定连接标识符，则使用上一个打开的连接。如果没有打开的连接，本函数将无参数调用 mysql_connect()函数来尝试打开一个并使用之。每个其后的 mysql_query()函数调用都会作用于活动数据库。

下面应用 mysql_connect()函数连接 MySQL 服务器，然后应用 mysql_select_db()函数连接 MySQL 数据库，代码如下：

```
<?php
$conn = mysql_connect("localhost", "root", "root");                          //连接 MySQL 服务器
$db=mysql_select_db("db_pursey", $conn) or die ("数据库连接失败: " .mysql_error()); //连接数据库 db _pursey
mysql_query("set names gb2312");                                             //采用 GB2312 编码方式
?>
```

3. mysql_query()函数

mysql_query()函数用来根据连接标识符向该数据库服务器的当前数据库发送查询。语法如下：





```
int mysql_query(string query ,int [link_identifier]);
```

其中，query 是查询字符串；link_identifier 是数据库连接标识符。

mysql_query()函数在执行成功时返回一个结果标识符，失败时返回 false。

4．date()函数

date()函数主要用于格式化一个本地时间/日期。语法如下：

```
string date ( string format , int timestamp)
```

该函数返回将参数 timestamp 按照指定格式格式化而产生的字符串，其中参数 timestamp 是可选的，默认值为 time()，即如果没有给出时间戳，则使用本地当前时间。

date()函数的参数 format 的格式化选项如表 1.6 所示。

表 1.6　参数 format 的格式化选项

参　数	说　明
a	小写的上午和下午值，返回值为 am 或 pm
A	大写的上午和下午值，返回值为 AM 或 PM
B	Swatch Internet 标准时，返回值为 000～999
d	月份中的第几天，有前导零的 2 位数字，返回值为 01～31
D	星期中的第几天，文本格式，3 个字母，返回值为 Mon～Sun
F	月份，完整的文本格式，返回值为 January～December
g	小时，12 小时格式，没有前导零，返回值为 1～12
G	小时，24 小时格式，没有前导零，返回值为 0～23
h	小时，12 小时格式，有前导零，返回值为 01～12
H	小时，24 小时格式，有前导零，返回值为 00～23
i	有前导零的分钟数，返回值为 00～59
I	判断是否为夏令时，返回值如果是夏令时为 1，否则为 0
j	月份中的第几天，没有前导零，返回值为 1～31
l	星期数，完整的文本格式，返回值为 Sunday～Saturday
L	判断是否为闰年，返回值如果是闰年为 1，否则为 0
m	数字表示的月份，有前导零，返回值为 01～12
M	3 个字母缩写表示的月份，返回值 Jan～Dec
n	数字表示的月份，没有前导零，返回值为 1～12
o	与格林威治时间相差的小时数，例如 0200
r	RFC 822 格式的日期，例如，Thu，21 Dec 2000 16：01：07 +0200
s	秒数，有前导零，返回值为 00～59
S	每月天数后面的英文后缀，2 个字符，例如 st、nd、rd 或者 th。可以和 j 一起使用
t	指定月份所应有的天数
T	本机所在的时区
U	从 UNIX 纪元（January 1 1970 00:00:00 GMT）开始至今的秒数
w	星期中的第几天，数字表示，返回值为 0～6
W	ISO-8601 格式年份中的第几周，每周从星期一开始

续表

参 数	说 明
Y	4 位数字完整表示的年份，返回值如 1998、2008
y	2 位数字表示的年份，返回值如 88 或 08
z	年份中的第几天，返回值为 0～366
Z	时差偏移量的秒数。UTC 西边的时区偏移量总是负的，UTC 东边的时区偏移量总是正的，返回值为-43200～43200

注意 有效的时间戳典型范围是格林威治时间 1901 年 12 月 13 日 20:45:54 到 2038 年 1 月 19 日 03:14:07（此范围符合 32 位有符号整数的最小值和最大值）。在 Windows 系统中此范围限制为从 1970 年 1 月 1 日到 2038 年 1 月 19 日。

例如，应用 date()函数格式化一个日期，并输出日期的值。代码如下：

```
<?php
echo date("y:m:d");
?>
```

结果为：07:07:11。

1.8.3 免费供求信息发布模块的实现过程

免费供求信息发布使用的数据表：tb_info

用户通过单击前台页面导航栏中的“我要发布”超链接，进入信息发布页面，如图 1.61 所示。程序会先验证用户是否输入信息，若验证失败，则返回信息发布页面，进行相应提示；若验证成功，则向数据库中插入记录，完成发布操作。



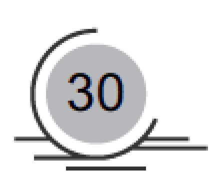
图 1.61 免费供求信息发布网页

在功能导航页 top.php 中，在“我要发布”图像上添加图像映射，并绘制热区。代码如下：

例程03 代码位置：光盘\TM\01\99pursey\top.php

```
<map name="Map">                                                   //图像映射
      <area shape="rect" coords="356,8,424,28" href="release.php">
</map>
```

在信息发布页面选择要发布的信息类型后，填写真实有效的供求信息。为了避免用户添加空信息，



在单击“发布信息”按钮时，应用 JavaScript 脚本自定义一个 checkform()函数，验证提交的表单各元素是否为空值，如果为空，则弹出提示信息，并将焦点定位到为空值的表单元素。代码如下：

例程04　代码位置：光盘\TM\01\99pursey\release_content.php

```
<script language="javascript">
function checkform(form){                              //自定义一个 JavaScript 函数 checkform()
    for(i=0;i<form.length;i++){                        //应用 for 循环语句检索 form 表单元素的值是否为空
        if(form.elements[i].value==""){                //如果 form 表单中某个元素值为空
            alert("请将发布信息填写完整！");              //弹出提示信息
            form.elements[i].focus();                  //将焦点的值定位到为空值的表单元素
            return false;                              //返回焦点
        }
    }
}
</script>
```

创建与数据库 db_pursey 的连接，代码如下：

例程05　代码位置：光盘\TM\01\99pursey\conn\conn.php

```
<?php
$link=mysql_connect("localhost","root","root");       //连接 MySQL 服务器
mysql_select_db("db_pursey ",$link);                   //连接 MySQL 数据库文件 db_pursey
mysql_query("set names gb2312");                       //采用 GB2312 编码方式
?>
```

代码贴士

❶ mysql_connect()函数：连接 MySQL 服务器，详细讲解参见 1.8.2 节。

❷ mysql_select_db()函数：连接 MySQL 数据库文件，详细讲解参见 1.8.2 节。

❸ mysql_query()函数：用来向数据库服务器发送编码方式，详细讲解参见 1.8.2 节。

提交表单信息到数据处理页，应用 insert…into 语句向免费供求信息表中添加供求信息。如果信息添加成功，则弹出成功的信息提示；否则弹出失败的提示信息。代码如下：

例程06　代码位置：光盘\TM\01\99pursey\release_ok.php

```
<?php
include("conn/conn.php");                              //连接数据库文件
$type=$_POST[type];                                    //获取信息类型
$title=$_POST[title];                                  //获取信息标题
$content=$_POST[content];                              //获取信息内容
$linkman=$_POST[linkman];                              //获取联系人
$tel=$_POST[tel];                                      //获取联系电话
❶   $edate=date("Y-m-d h:i:s");                        //获取发布时间
❷   $sql=mysql_query("insert into tb_info(type,title,content,linkman,tel,checkstate,edate)
values('$type','$title','$content','$linkman','$tel',0,'$edate')"); //将免费的供求信息添加到数据表中
if($sql){                                              //如果添加操作成功，则弹出提示信息
    echo "<script>alert('恭喜您，信息发布成功！');window.location.href='release.php';</script>";
}else{                                                 //如果添加操作失败，则弹出提示信息
    echo "<script>alert('对不起，信息发布失败！');history.back();</script>";
}
?>
```

代码贴士

❶ date()函数：格式化一个本地时间/日期，详细讲解参见 1.8.2 节。

❷ insert into：向指定的数据表中添加数据信息。

1.9 信息检索模块设计

1.9.1 信息检索模块概述

信息检索是对已存在于数据库中的数据按条件进行筛选浏览，是查看历史信息和确认数据操作最为快速、有效的办法。信息检索模块主要通过选择信息类型和输入查询关键字模糊查询供求信息资源，并输出查询结果。考虑到供求信息的信息量较大，因此本模块对与查询关键字相匹配的查询结果进行描红，从而方便用户的浏览。信息检索模块的示意图如图 1.62 所示。

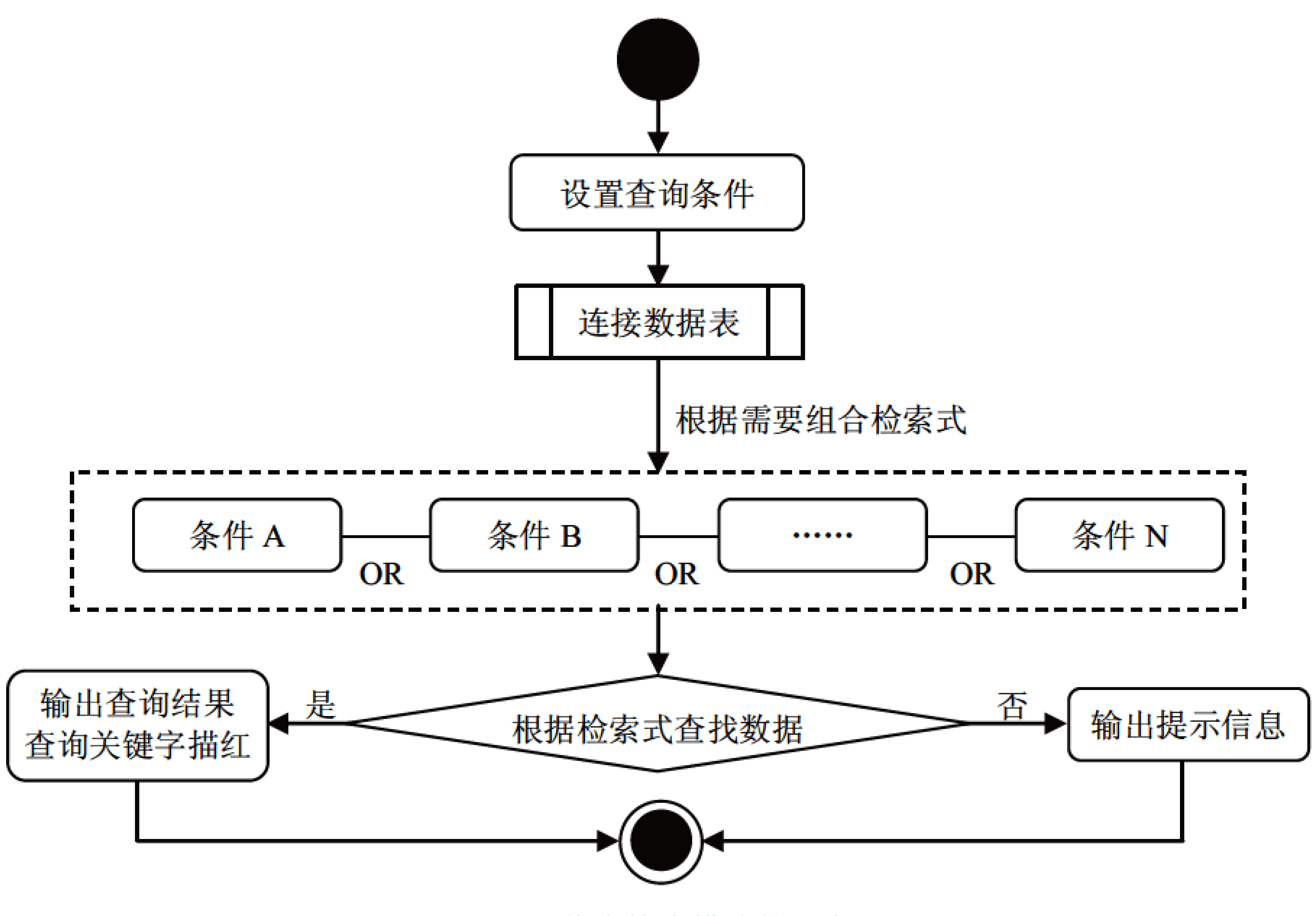
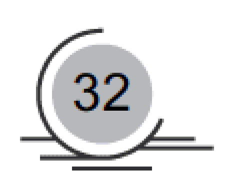


图 1.62　信息检索模块的示意图

1.9.2 信息检索模块技术分析

在对数据进行查询后，最终需要将查询结果显示在页面中反馈给浏览者。在 PHP 中，查询结果的显示方式有很多种，最常用的就是表格显示方式。因为采用这种方式显示的数据条理清晰、简洁明了。

在利用表格显示查询结果时，通常是将查询结果保存在结果集中，然后需要使用 do…while 循环将其查询结果显示出来。需要注意的是，需要先判断查询结果是否为空，只有查询结果不为空时，才可以使用循环语句显示数据。为了使读者更好地理解通过表格显示查询结果，下面给出其实现流程图，如图 1.63 所示。





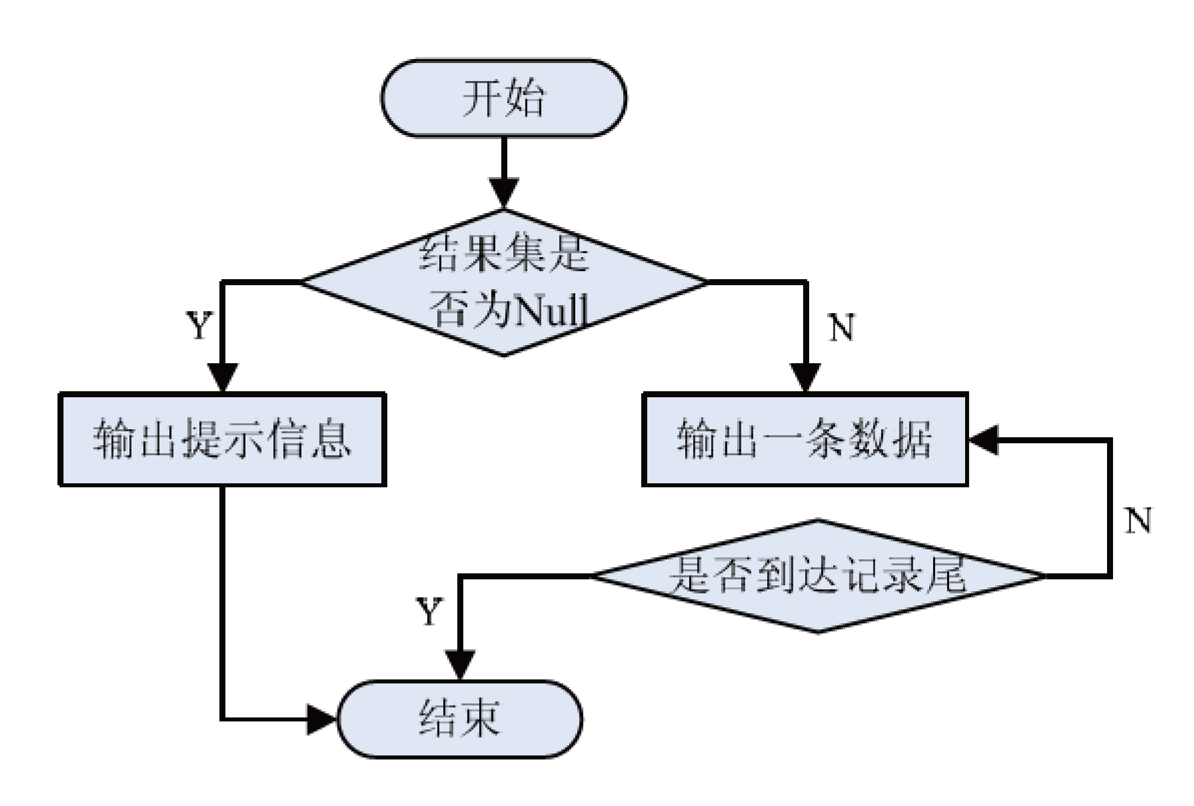


图 1.63　信息检索模块流程图

考虑到用户不可能全面了解数据表中的数据信息，例如不能确定所要查询信息的内容、查询的主题等，这时就需要使用 like 进行模糊查询。like 关键字需要使用通配符在字符串内查找指定的模式，所以读者需要了解通配符及其含义。通配符的含义如表 1.7 所示。

表 1.7　like 关键字中的通配符及说明

通　配　符	说　　明
%	由零个或更多字符组成的任意字符串
_	任意单个字符
[ ]	用于指定范围，例如[A～F]表示 A～F 范围内的任何单个字符
[^]	表示指定范围之外的，例如[^ A～F]表示 A～F 范围以外的任何单个字符

如果想查询包含“女子公寓”的信息，可以使用 like 运算符配合通配符“%”完成。其 SQL 语句如下：

```
select * from tb_info where content like '%女子公寓%';
```

如果想查找信息类型为“公寓信息”或者内容为“女子公寓”的信息时配合 or 运算符来使用。其 SQL 语句如下：

```
select * from tb_info   where type='公寓信息' or content like'%女子公寓%'
```

本模块实现付费信息与查询关键字相匹配的信息的 SQL 语句如下：

```
$type=$_POST[type];                                          //获取信息类型
$content=$_POST[content];                                    //获取查询关键字
$sql1=mysql_query("select * from tb_leaguerinfo  where checkstate=1 and type='$type' and content like'%$content%' or title like'%$content%' or linkman like'%$content%' or tel like'%$content%'");
$info1=mysql_fetch_array($sql1);                             //检索付费的信息
```

本模块实现免费信息与查询关键字相匹配的信息的 SQL 语句如下：

```
$sql=mysql_query("select * from tb_info  where checkstate=1 and type='$type' and content like'%$content%' or title like'%$content%' or linkman like'%$content%' or tel like'%$content%'");
$info=mysql_fetch_array($sql);                               //检索免费的信息
```



注意 当满足数据表中多个字段中的任一字段时，可以使用 or 运算符将多个条件连接。

另外，由于搜索的内容中文字比较多，为了方便浏览者查找自己所关注的内容信息，所以在搜索引擎中加入了描红功能。描红功能主要用 str_ireplace()函数实现，该函数的具体讲解读者可参见本章的 1.14.1 节。

1.9.3 信息检索模块的实现过程

信息检索模块使用的数据表：tb_info、tb_ leaguerinfo

在开发信息检索模块时，由于该网站含有大量的数据信息，为了方便用户浏览网站信息，需要添加复合条件查询实现搜索功能。在信息检索区的“关键字”文本框中输入欲查询的关键字，在“条件”下拉列表框中选择要搜索的信息类型，然后单击“开始搜索”按钮，对指定条件的记录进行检索并输出结果集到浏览器，同时为了方便浏览者查找自己所关注的内容信息，本模块对查询关键字进行描红。运行结果如图 1.64 所示。



图 1.64 信息检索页面的运行结果

信息检索页面中所涉及的重要表单元素如表 1.8 所示。

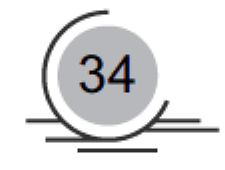
表 1.8 信息检索页面所涉及的重要表单元素

名 称	元 素 类 型	重 要 属 性	含 义
form1	form	method="post" action="findinfo.php"	表单
content	text	id="content" size="20"	查询关键字
type	select	<select name="type"> <option value="招聘信息">-招聘信息-</option> <option value="求职信息" selected>-求职信息-</option> </select>	信息类型
search	image	src="Images/btn1.gif" onClick="return chkinput(form)"	开始搜索按钮

应用 JavaScript 脚本自定义一个 chkinput()函数，实现对表单提交的信息进行验证。代码如下：

例程07 代码位置：光盘\TM\01\99pursey\left.php

```
<script language="javascript">
        function chkinput(form){                                    //自定义一个 chkinput()函数
            if(form.content.value==""){                             //判断如果查询关键字文本框等于空
                alert("请输入查询关键字!");                         //则弹出提示信息
                form.content.select();                              //重新定位焦点
                return false;                                       //返回表单元素
            }
        }
</script>
```





将表单信息提交到数据处理页，连接数据库文件，接收表单信息，然后用 mysql_query()函数向服务器发送 SQL 语句，检索与查询关键字相匹配的信息资源。代码如下：

例程08　代码位置：光盘\TM\01\99pursey\findinfo.php

```
<?php
include("conn/conn.php");                                //连接数据库文件
$type=$_POST[type];                                      //获取信息类型
$content=$_POST[content];                                //获取查询关键字
$sql1=mysql_query("select * from tb_leaguerinfo where checkstate=1 and type='$type' and content like'%$content%'
or title like'%$content%' or linkman like'%$content%' or tel like'%$content%'");
$info1=mysql_fetch_array($sql1);                         //检索付费的供求信息
$sql=mysql_query("select * from tb_info   where checkstate=1 and type='$type' and content like'%$content%' or
title like'%$content%' or linkman like'%$content%' or tel like'%$content%'");
$info=mysql_fetch_array($sql);                           //检索免费的供求信息
?>
```

说明　信息检索需要从免费供求信息表 tb_info 和付费供求信息表 tb_leaguerinfo 中获取数据，因此需要向 MySQL 服务器传递两条 SQL 语句。

用 do…while 循环语句输出付费信息与查询关键字相匹配的信息资源，并用 str_ireplace()函数对查询关键字实现描红功能。代码如下：

例程09　代码位置：光盘\TM\01\99pursey\findinfo.php

```
<!--下面输出的是付费信息与查询关键字相匹配的信息-->
<?php
if($info1){                                              //如果检索到了付费信息
     do{                                                 //则用 do…while 循环语句输出付费信息
 ?>
    <table width="540" border="0" cellspacing="0" cellpadding="0">
       <tr>
         <td height="26">
         <!--应用 str_ireplace()函数对查询关键字进行描红-->
         <!--对与查询关键字所匹配的信息类型进行描红-->
         『<?php echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[type]);?>』nbsp;
         <!--对与查询关键字所匹配的信息标题进行描红-->
         <?php echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[title]);?>  
         <!--对与查询关键字所匹配的发布时间进行描红-->
         <?php echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[edate]);?></td>
        </tr>
        <tr>
         <td height="26">    
         <!--对与查询关键字所匹配的信息内容进行描红-->
         <?php echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[content]);?></td>
        </tr>
        <tr>
         <td height="26"> 联系人：
         <!--对与查询关键字所匹配的联系人进行描红-->
         <?php echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>", $info1[linkman]); ?> 
```

```
            <!--对与查询关键字所匹配的联系电话进行描红-->
            联系电话：<?php echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[tel]);?>
            <!-- ------------------------------------------------------------- -->
            </td>
        </tr>
    </table>
<?php
    }while($info1=mysql_fetch_array($sql1));                    //循环语句结束
?>
```

说明 免费信息的输出方式与付费信息的基本类似，代码部分略，详见本书附赠光盘。

免费信息的输出方式与付费信息的基本类似，下面给出实现过程的核心代码结构。

例程10　代码位置：光盘\TM\01\99pursey\findinfo.php

```
<!-- -----------------------------下面输出的是免费信息与查询关键字相匹配的信息---------------------------- -->
<?php
if($info){                                                          //如果检索到了免费信息
    do{                                                             //则用 do…while 循环语句输出付费信息
?>
    …                                                               //免费信息的输出方式与付费信息的类似，代码略
<?php
    } while($info=mysql_fetch_array($sql));                         //do…while 循环语句结束
  }                                                                 //if 条件语句结束
?>
```

如果在免费供求信息表和付费供求信息表中没有检索到与查询关键字相匹配的数据，则弹出提示信息。代码如下：

例程11　代码位置：光盘\TM\01\99pursey\findinfo.php

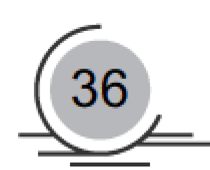
```
<table width="540" border="0" cellspacing="0" cellpadding="0">
    <tr>
      <td align="center">您检索的信息资源不存在！</td>
    </tr>
</table>
```

1.10　后台首页设计

1.10.1　后台首页概述

程序开发人员在设计网站后台首页时，主要从后台管理人员对功能的易操作性、实用性、网站的易维护性考虑，因此采用了框架技术。九九度供求信息网后台首页主要包含以下内容：

☑ 发布付费的供求信息（包括公寓信息、招聘信息、求职信息、培训信息、家教信息、房屋信息、车辆信息、求购信息、出售信息、招商引资、寻人/物启示等），以及付费信息的浏览、审核及删除功能。



☑　免费信息的浏览、审核及删除功能。

☑　企业广告信息的发布、浏览、前台推荐显示、删除功能。

☑　“网站首页”超链接，为管理员进入前台提供一个入口。

☑　“退出登录”超链接，用于注销当前用户。

本案例中提供的后台首页如图 1.65 所示。该页面在本书光盘中的路径为\TM\01\99pursey\admin\index.php。

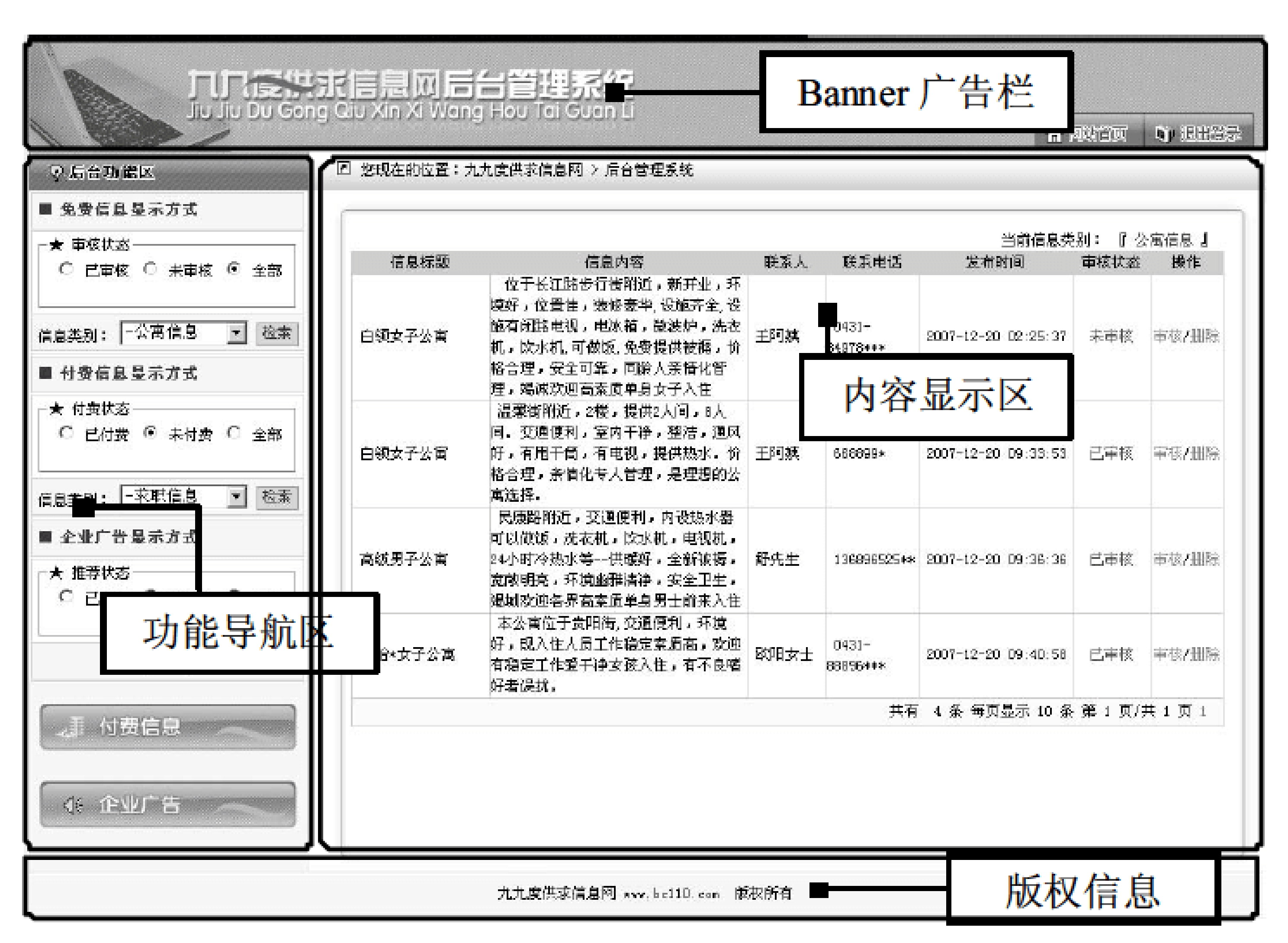


图 1.65　供求信息网站后台首页

1.10.2　后台首页技术分析

九九度供求信息网后台采用框架技术进行页面布局。所谓框架就是网页的各部分为相互独立的网页，又由一个网页将这些分开的网页组成一个完整的网页，显示在浏览者的浏览器中，重复出现的内容被固定下来，每次浏览者发出对页面的请求时，只下载发生变化的框架页面，其他子页面保持不变。

使用框架可以将容器窗口划分为若干个子窗口，在每个子窗口可以分别显示不同的网页。首先在 Dreamweaver 8 中创建一个“左→中→右”的框架集，然后在标识①中添加“上→中→下”的框架集，最后在标识②中添加一个“左→右”的框架集，从而完成一个完整的后台框架。构建框架的流程如图 1.66 所示。

图 1.66　网站后台框架流程

使用框架可以非常方便地完成导航工作。下面详细介绍框架网页的基本结构、设置框架集的属性和设置框架的属性。

1．框架网页的基本结构

框架网页通过一个或多个 frameset 和 frame 标记来定义。在框架网页中，将 frameset 标记置于 head 标记之后，以取代 body 标记的位置，当客户端浏览器不支持框架网页时，还可以使用 noframes 标记给

出框架不能被显示时的替换内容。框架网页的基本结构如下：

```
<html>
<head>
<title>基本框架页</title>
</head>
<frameset>
      <frame>
      <frame>
</frameset>
<noframes>
       <body>
       对不起！您的浏览器不支持框架页面的显示！
       </body>
</noframes>
</html>
```

2. 设置框架集的属性

框架集包含如何组织各个框架的信息，可以通过 frameset 标记来定义。框架是按照行和列来组织的，可以使用 frameset 标记的下列属性对框架的结构进行设置。

（1）左右分割窗口属性 cols

在水平方向上将浏览器分割成多个窗口，可以通过框架的左右分割窗口属性 cols 实现。其语法格式如下：

```
<frameset cols="value,value,…">
   <frame>
   <frame>
</frameset>
```

其中，value 用于指定各个框架的列宽，取值有像素、百分比（%）和相对尺寸（*）3 种形式。

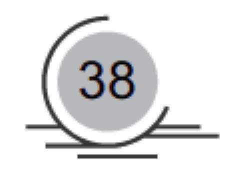
例如，若要通过框架将浏览器窗口划分为 3 列，其中第 1 列占浏览器窗口宽度的 20%，第 2 列为 120 像素，第 3 列为浏览器窗口剩余部分的框架。代码如下：

```
<frameset cols="20%,120,*" >
   <frame>
   <frame>
</frameset>
```

技巧 如果将 cols 属性设置为“*，*，*”，则表示将窗口划分成 3 个等宽的框架；如果将 cols 属性设置为“*，2*，3*”，则表示左边的框架占窗口宽度的 1/6，中间的框架占窗口宽度的 1/3，右边的框架占窗口宽度的 1/2。

（2）上下分割窗口属性 rows

在垂直方向上将浏览器分割成多个窗口，可以通过框架的上下分割窗口属性 rows 实现。其语法格式如下：



```
<frameset rows="value,value,…">
  <frame>
  <frame>
</frameset>
```

其中 value 用于指定各个框架的行高，取值有像素、百分比（%）和相对尺寸（*）3 种形式，设置方法与 COLS 属性类似。例如，若要通过框架将浏览器窗口划分为 3 行，其中的第 1 行占浏览器窗口宽度的 20%，第 2 行为 120 像素，第 3 行为浏览器窗口剩余部分的框架。代码如下：

```
<frameset rows="20%,120,*" >
  <frame>
  <frame>
</frameset>
```

（3）框架边框显示属性 frameborder

该属性用于指定框架周围是否显示边框，取值为 1（显示边框，默认值）或 0（不显示边框）。

（4）framespacing

该属性用于指定框架之间的间隔，以像素为单位。如果不设置该属性，则框架之间没有间隔。

（5）指定边框宽度属性 border

该属性用于指定边框的宽度，只有 frameborder 属性为 1 时有效。

3．设置框架的属性

使用 FRAME 标记可以设置框架的属性，包括框架的名称、框架是否包含滚动条以及在框架中显示的网页等。其语法格式如下：

```
<frame name="框架名称" src="文件" frameborder="数值" scrolling="值" [noresize] >
```

属性说明如下。

- ☑ name：指定框架的名称。
- ☑ src：指定在框架中显示的网页文件（包括 HTML、ASP 等网页文件）。
- ☑ frameborder：指定框架周围是否显示边框，取值为 1（显示）或 0（不显示）。默认值为 1。
- ☑ scrolling：指定框架是否包含滚动条。如果将该属性设置为 yes，则框架包含滚动条；若将该属性设置为 no，则框架不包含滚动条；如果将该属性设置为 auto，则在需要时包含滚动条。
- ☑ noresize：可选属性，若指定了该属性，则不能调整框架的大小。

1.10.3　后台首页的实现过程

根据 1.10.1 节和 1.10.2 节的页面概述及实现技术分析，需要分别创建实现各区域的 PHP 文件，如实现 Banner 广告栏的 top.php、功能导航栏的 left.php、内容显示区的 main.php 和页尾文件 bottom.php 等。实现该系统后台框架布局的完整代码如下：

例程12　代码位置：光盘\TM\01\99pursey\admin\index.php

```
<frameset rows="*" cols="1*,1005,1*" framespacing="0" frameborder="NO" border="0">
  <frame src="blank.php" name="left" scrolling="NO" noresize>                    <!--设置空框架页-->
```

```
<frameset rows="1005,*" cols="*" framespacing="0" frameborder="NO" border="0">
        <frameset rows="94,*,190" cols="*" framespacing="0" frameborder="NO" border="0">
          <frame src="top.php" name="topFrame" scrolling="NO" noresize>
              <frameset   rows="*" cols="229,*"   framespacing="0" frameborder="NO" border="0">
                  <frame src="left.php" name="leftFrame" scrolling="NO" noresize>
                  <frame src="main.php" name="mainFrame" scrolling="NO" noresize>
             </frameset>
          <frame src="bottom.php" name="bottomFrame" scrolling="NO" noresize>
        </frameset>
        <frame src="blank.php" name="right" scrolling="NO" noresize>
    </frameset>
<frame src="blank.php"></frameset>                                            <!--设置空框架页-->
<noframes><body>
</body></noframes>
```

注意 在建设 Web 网站时，如何让不同分辨率的用户都能看到网页的最佳效果是程序员在设计之初所要考虑的首要问题。为了使屏幕的分辨率在大于 1024×768 像素的设置时仍然处于居中显示，只需要在设置框架布局时，在主框架两侧各设置一个宽度相同的 blank.php 空页即可。

1.11 付费供求信息发布模块设计

1.11.1 付费供求信息发布模块概述

付费供求信息的发布提供对象为供求信息用户，是供求信息网站非常重要的功能，也是供求信息网站的盈利点。企业或用户可以根据自身需要对供求信息先进行付费，付费后由管理员在后台将供求信息发布到相应的信息类别中（共包括 11 个信息类别：招聘信息、求职信息、培训信息、公寓信息、家教信息、车辆信息、物品求购、物品出售、求兑出兑、寻求合作、企业广告等类型供求信息）。供求信息成功发布后，管理员需要在后台对发布的供求信息进行审核，如果审核通过后，则显示在前台相应的信息类别网页中。付费供求信息发布的流程如图 1.67 所示。

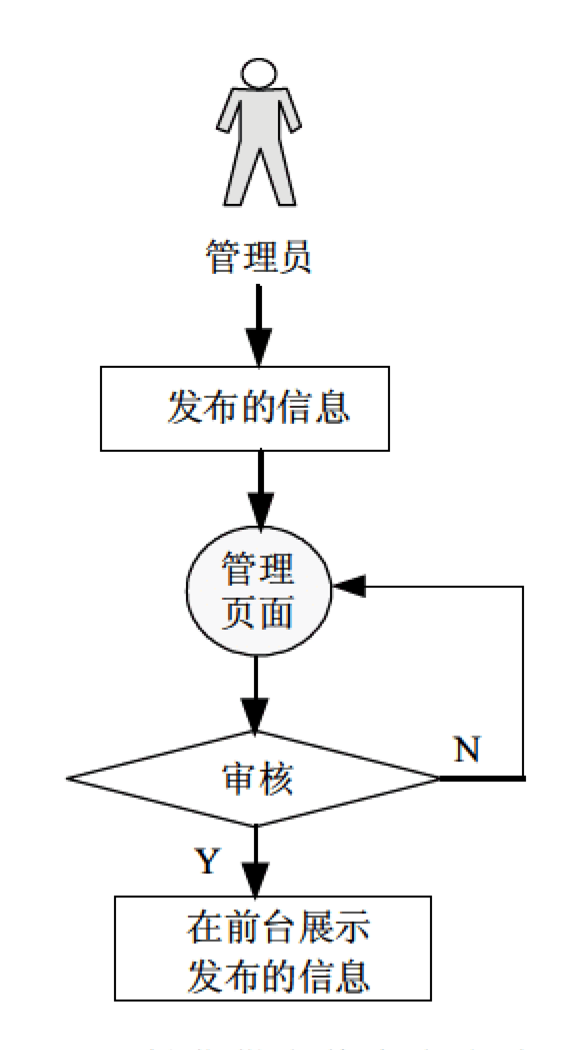
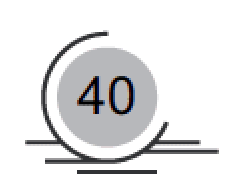


图 1.67　付费供求信息发布流程图

1.11.2 付费供求信息发布模块技术分析

付费供求信息与免费供求信息不同的是，付费供求信息不仅需要收取一定的费用，而且还需要一定的时间限制，例如，网站要求一个月（按 30 天计算）每条信息交 10 元的信息费，如果用户交纳 20 元，那么信息显示的天数就是 60 天。在前台进行显示时，不需要管理员进行手动管理，而是通过程序直接计算出信息显示的截止时间。





信息显示的截止时间=“系统当前日期”+“信息的有效天数（与用户交纳的信息费相关）”。

自动计算信息显示的截止时间的具体代码如下：

```
$days=$_POST[days];                                    //通过表单传值获取信息显示的天数
$showday=date("Y-m-d",(time()+3600*24*$days));         //信息显示的截止时间
```

说明 信息的有效天数与用户交纳的信息费相关，交费不通过本程序完成，因此，信息的有效天数需要管理员手动添加。

1.11.3　付费供求信息发布模块的实现过程

付费供求信息发布使用的数据表：tb_leaguerinfo

用户通过单击页面导航区的“付费信息”超链接，进入付费信息发布页面，如图 1.68 所示。填写真实有效的付费信息，单击“发布信息”按钮，程序会先验证用户输入的信息，若验证失败，则返回信息发布页面，进行相应提示；若验证成功，则向数据库中插入记录，完成付费信息的发布操作。



图 1.68　付费供求信息发布页面运行结果

在左侧框架 left.php 页中，添加“付费信息”图像域及表单。代码如下：

例程13　代码位置：光盘\TM\01\99pursey\admin\left.php

```
<form name="form1" method="post" action="release_content.php" target="mainFrame">
    <input name="imageField" type="image" class="input1" src="images/btn_fufei.gif" width="210" height="39" border="0">
</form>
```

单击“付费信息”按钮，将信息页 release_content.php 中的内容显示在框架显示页 mainFrame 中。

付费供求信息发布页面主要用于发布付费的供求信息，该页面中所涉及的重要表单元素如表 1.9 所示。

表 1.9　付费供求信息页面所涉及的重要表单元素

名　称	元素类型	重要属性	含　义
form1	form	method="post" action="release_ok.php"	表单
type	select	<select name="type"> <option value="招聘信息">-招聘信息-</option> <option value="求职信息" selected>-求职信息-</option> … <option value="寻人/物启示">-寻人/物启示-</option> </select>	信息类型
flag	checkbox	class="input1"　value="1" checked	“是否付费”复选框
title	text	size="50"	信息标题
content	textarea	cols="55" rows="8"	信息内容
linkman	text	size="30"	联系人
tel	text	size="30"	联系电话
days	text		有效天数
imageField	image	src="images/fa.jpg" onClick="return checkform(form);"	“发布信息”按钮

在付费信息发布页面选择要发布的信息类型后，填写真实有效的供求信息。为了避免用户添加空信息，在单击“发布信息”按钮时，应用 JavaScript 脚本自定义一个 checkform()函数，验证提交的表单各元素是否为空值，如果为空，则弹出提示信息，并将焦点定位到为空值的表单元素。checkform()函数的代码部分与例程 04 相同，这里不再赘述。

提交表单信息到数据处理页，应用 insert…into 语句向付费供求信息表中添加供求信息。如果信息添加成功，则弹出成功的信息提示；否则弹出失败的提示信息。代码如下：

例程 14　代码位置：光盘\TM\01\99pursey\admin\release_ok.php

```
<?php
include("../conn/conn.php");                                      //连接数据库文件
$type=$_POST[type];                                               //获取信息类型
$flag=$_POST[flag];                                               //获取付款状态
$title=$_POST[title];                                             //获取信息标题
$content=$_POST[content];                                         //获取信息内容
$linkman=$_POST[linkman];                                         //获取联系人
$days=$_POST[days];                                               //获取发布时间
$tel=$_POST[tel];                                                 //获取联系电话
$sdate=date("Y-m-d");                                             //当前系统时间
❶    $showday=date("Y-m-d",(time()+3600*24*$days));               //获取信息的有效时间
$sql=mysql_query("insert into tb_leaguerinfo(type,title,content,linkman,tel,sdate,showday,checkstate)
values('$type','$title','$content','$linkman','$tel','$sdate','$showday',$flag)");  //将付费的供求信息添加到数据表中
if($sql){                                                         //如果添加操作成功，则弹出提示信息
❷    echo "<script>alert('信息发布成功！'); parent.mainFrame.location.href='release_content.php';</script>";
}else{                                                            //如果添加操作失败，则弹出提示信息
     echo "<script>alert('信息发布失败！ ');history.back();</script>";
```



```
}
?>
```

代码贴士

❶ date("Y-m-d",(time()+3600*24*$days)): 信息的有效时间=当前期日期+付费期限。应用 time()函数获取当前日期时间戳，付费期限的时间戳等于 3600 秒 × 24 小时 × 指定天数，并通过 date()函数格式化为指定日期格式。

❷ parent.mainFrame.location.href='release_content.php': 刷新父框架页 release_content.php 中的信息。

1.12　付费信息管理模块设计

1.12.1　付费信息管理模块概述

付费信息管理模块主要包括付费信息列表、付费信息审核和付费信息删除等 3 个功能，其框架如图 1.69 所示。

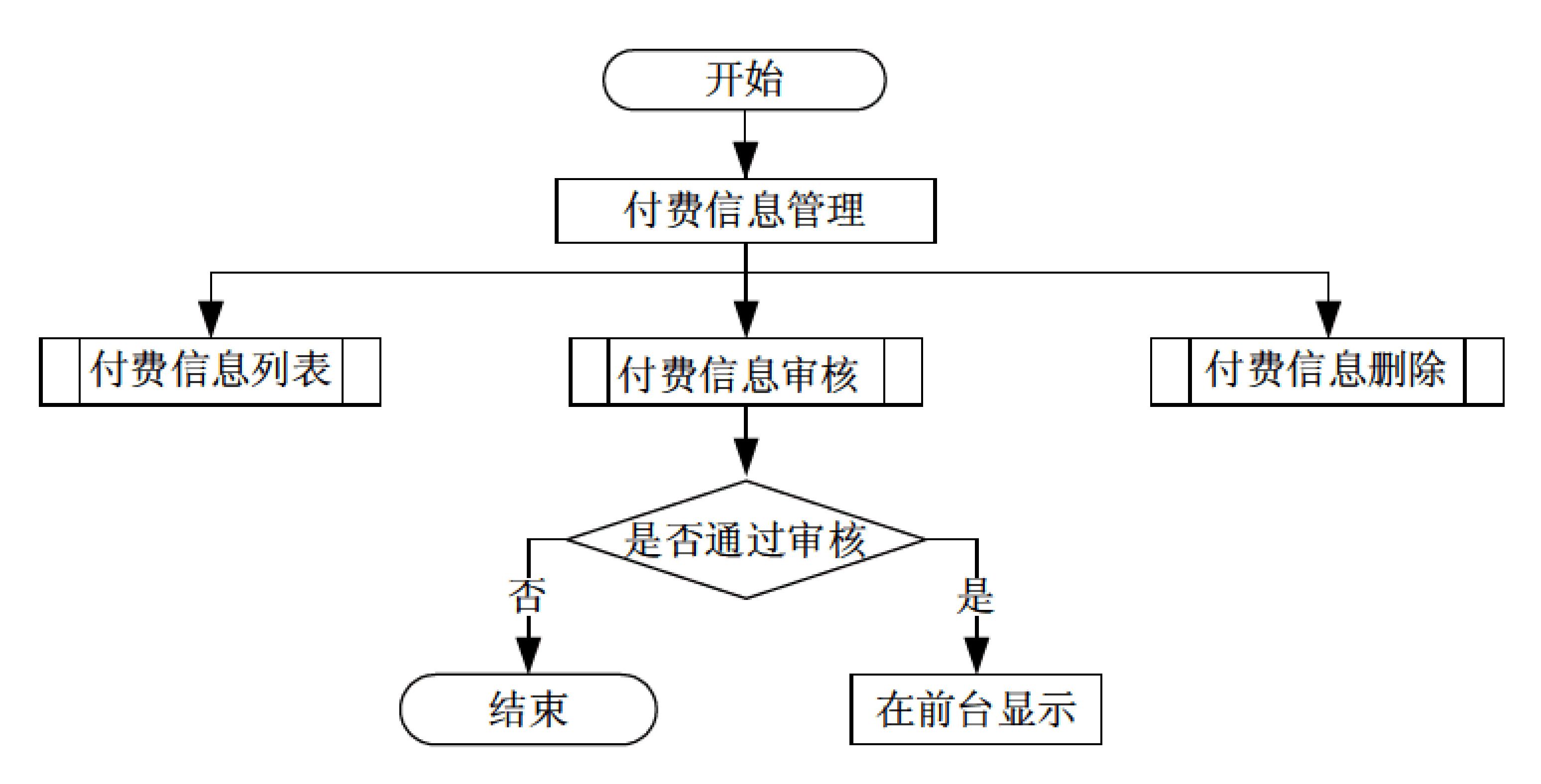


图 1.69　付费信息管理模块的框架图

1.12.2　付费信息管理模块技术分析

付费信息管理页面在实现信息审核及删除的功能时应用到了 UPDATE 更新语句和 DELETE 删除语句。下面对这两个语句进行详细的讲解。

1．UPDATE 语句

UPDATE 语句用来改变单行上的一列或多列的值，或者改变单个表中选定的一些行上的多个列值。UPDATE 语句的语法如下：

```
UPDATE<table_name | view_name>
SET <column_name>=<expression>
    [...,<last column_name>=<last expression>]
[WHERE<search_condition>]
```

UPDATE 语句的参数说明如表 1.10 所示。

表 1.10　UPDATE 语句的参数说明

参　　数	说　　明
table_name	需要更新的表的名称。如果该表不在当前服务器或数据库中，或不为当前用户所有，这个名称可用链接服务器、数据库和所有者名称来限定
view_name	要更新的视图的名称。通过 view_name 来引用的视图必须是可更新的。用 UPDATE 语句进行的修改，至多只能影响视图的 FROM 子句所引用的基表中的一个
SET	指定要更新的列或变量名称的列表
column_name	含有要更改数据的列的名称。column_name 必须驻留于 UPDATE 子句中所指定的表或视图中。标识列不能进行更新。如果指定了限定的列名称，限定符必须同 UPDATE 子句中的表或视图的名称相匹配
expression	变量、字面值、表达式或加上括号返回单个值的 subSELECT 语句。expression 返回的值将替换 column_name 或@variable 中的现有值
WHERE	指定条件来限定所更新的行。根据所使用的 WHERE 子句的形式，有两种更新形式： ☑ 搜索更新指定搜索条件来限定要删除的行 ☑ 定位更新使用 CURRENT OF 子句指定游标。更新操作发生在游标的当前位置
<search_condition>	为要更新行指定需满足的条件。搜索条件也可以是联接所基于的条件。对搜索条件中可以包含的谓词数量没有限制

注意　一定要确保不要忽略 WHERE 子句，除非想要更新表中的所有行。

下面应用 UPDATE 语句将指定职员的工资进行调整，例如，将“小璇”的基本工资由 2600 元修改为 3000 元。其 SQL 语句如下：

```
update tab_laborage set jbgz=3000 where name='小璇'
```

2. DELETE 语句

DELETE 语句实现删除数据记录。DELETE 语句的语法如下：

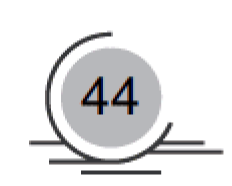
```
DELETE FROM <table_name >
[WHERE<search-condition>]
```

DELETE 语句的参数说明如表 1.11 所示。

表 1.11　DELETE 语句的参数说明

参　　数	说　　明
FROM	可选的关键字，可用在 DELETE 关键字与目标 table_name、view_name 或 rowset_function_limited 之间
table_name	是要从其中删除行的表的名称
<search_condition>	指定删除行的限定条件。对搜索条件中可以包含的谓词数量没有限制

下面应用 DELETE 语句删除“职员姓名=小璇”的员工基本信息。其 SQL 语句如下：



```
DELETE tab_staffer WHERE ygname='小璇'
```

1.12.3　付费信息显示的实现过程

付费信息显示页面使用的数据表：tb_leaguerinfo

管理员在后台功能导航区的“付费信息显示方式”栏中选择相应的信息类别，然后按“已付费”、“未付费”或“全部”中的任意一种状态对付费信息进行管理。例如，在“信息类别”下拉列表框中选择“公寓信息”，在“付费状态”选项组中选中“未付费”单选按钮，单击“检索”按钮提交表单，程序将按指定条件显示出符合条件的所有信息，运行结果如图 1.70 所示。



图 1.70　付费信息显示页面的运行结果

本系统提供了一组单选按钮组成的“付费状态”选项组，分为已付费、未付费和全部 3 个选项。选中“未付费”单选按钮，则传递的值为 0；选中“已付费”单选按钮，则传递的值为 1；选中“全部”单选按钮，则传递的值为 all。还提供了一个下拉列表框，供用户选择信息类别。将这些单选按钮与下拉列表框都在一个表单中实现，这样，当单击“检索”按钮提交表单后，选择的状态会通过表单进行传递。其表单代码如下：

例程15　代码位置：光盘\TM\01\99pursey\admin\left.php

```
❶ <form name="form3" method="post" action="find_mianfei.php" target="mainFrame">
    <tr>
      <td height="65" align="center">
❷   <fieldset style="height:60;width:210">
              <legend>★ 审核状态</legend>
                <input name="state" type="radio" class="input1" value="1">已审核
                <input name="state" type="radio" class="input1" value="0" checked>未审核
                <input name="state" type="radio" class="input1" value="all"> 全部
         </fieldset>
    </td>
    </tr>
    <tr>
    <td height="34" align="center" background="images/info_d.gif">信息类别：
         <select name="type">
           <option value="招聘信息">-招聘信息</option>
           <option value="求职信息" selected>-求职信息</option>
               …
         </select>
          <input type="submit" name="Submit" value="检索">
    </td>
      </tr>
    </form>
```

代码贴士

❶ target：指定链接的目标窗口，mainFrame 为内容显示区的框架名称。另外，target 有 4 个选项值，分别介绍如下。

☑ _blank：指定将链接的目标文件加载到未命名的新浏览器窗口中。

☑ _parent：指定将链接的目标文件加载到包含链接的父框架页或窗口中，如果包含链接的框不是嵌套的，则链接的目标文件加载到整个浏览器窗口中。

☑ _self：指定将链接的目标文件加载到链接所在的同一框架或窗口中。

☑ _top：指定将链接的目标文件加载到整个浏览器窗口中，并由此删除所有框架。

❷ <fieldset><legend>…</legend></fieldset>标签：在字符集包含的文本和其他元素外面绘制一个方框。该元素是块元素，必须成对出现。需要注意的是，fieldset 必须用在 form 表单中，一个表单可以有多个<fieldset>…</fieldset>，每对<fieldset>…</fieldset>为一组，每组的内容描述使用<legend>设置标题名称。

提交表单信息到 find_mianfei.php 页，程序将按管理员选择的指定条件显示出符合条件的所有信息。如果管理员选中“全部”单选按钮，那么系统的代码如下：

例程16 代码位置：光盘\TM\01\99pursey\admin\find_fufei.php

```
<?php
include("../conn/conn.php");                              //连接数据库文件
$state=$_POST[payfor];                                    //获取管理员选择的付费状态
$type=$_POST[select];                                     //获取管理员选择的信息类别

//如果管理员选择的付费状态为“全部”，则查询管理员指定信息类别下的所有付费信息，并按 id 降序排列
if($state=="all"){
    $sql=mysql_query("select *  from tb_leaguerinfo where type='$type' order by id");
}
else{
//如果管理员选择的付费状态为“已付费”或“未付费”状态，则按管理员指定的条件进行查询，并按 id 降序排列
    $sql=mysql_query("select *  from tb_leaguerinfo  where type='$type' and checkstate=$state order by id");
}
$info=mysql_fetch_array($sql);                            //执行 SQL 语句
?>
…                                                         //省略供求信息标题的 HTML 代码部分
<?php
if($info){                                                //如果检索到了查询记录
    do{                                                   //则应用 do…while 循环语句输出付费信息
        if($info[checkstate]==1){                         //如果付费状态的值为 1
            $state1="已付费";                             //则将“已付费”赋给变量$state1
        }else{                                            //如果付费状态的值为 0
            $state1="未付费";                             //则将“未付费”赋给变量$state1
        }
?>
<!-- -------------------------输出指定符合查询条件的付费信息----------------------------- -->
<tr bgcolor="#FFFFFF">
  <td> <?php echo $info[title];?></td>
  <td width="204"> <?php echo $info[content];?></td>
  <td> <?php echo $info[linkman];?></td>
  <td> <?php echo $info[tel];?></td>
  <td> <?php echo $info[sdate];?></td>
  <td> <?php echo $info[showday];?></td>
```

```
  <td align="center" class="style11"><?php echo $state1;?></td>
  <td align="center" bgcolor="#FFFFFF">
<a href="statefu_ok.php?id=<?php echo $info[id];?>&type=<?php echo $type;?>&state=<?php echo $state;?>">
审核</a>
/<a href="fudel_ok.php?id=<?php echo $info[id];?>&type=<?php echo $type;?>&state=<?php echo $state;?>">
删除</a>
</td>
</tr>
<!-- ---------------------------------------------------------------------------------------------------- -->
<?php
     }while($info=mysql_fetch_array($sql));                    //do…while 循环语句结束
}                                                              //if 条件语句结束

//如果未检索到与管理员指定条件相匹配的记录，则输出相关的提示信息
else{
?>
<tr align="center" bgcolor="#FFFFFF">
     <td colspan="8">对不起，您检索的信息不存在！</td>
  </tr>
<?php
}
?>
```

说明　限于篇幅，上面代码中省略了分页显示的代码部分，具体参见本书附赠光盘。

1.12.4　付费信息审核的实现过程

付费信息审核使用的数据表：tb_leaguerinfo

经过审核的信息说明该信息为已付款信息。如果企业或个人用户已登录供求信息但未直接付费，想要后期付款，那么管理员进行审核时，信息不能通过，要求必须进行付款，经过审核的信息才能在前台进行显示。“审核”超链接的代码如下：

例程17　代码位置：光盘\TM\01\99pursey\admin\find_fufei.php

```
<a href="statefu_ok.php?id=<?php echo $info[id];?>&type=<?php echo $type;?>&state=<?php echo $state;?>">
审核</a>
```

管理员单击对应主题信息后面的“审核”超链接，将信息所对应的 id 值、信息类型及审核状态传递到数据处理页 statefu_ok.php，用 UPDATE 语句将付费状态设置为 1，说明该信息已经付款。数据处理页的代码如下：

例程18　代码位置：光盘\TM\01\99pursey\admin\statefu_ok.php

```
<?php
include("../conn/conn.php");                                   //连接数据库文件
$id=$_GET[id];                                                 //获取信息 id 的值
$type=$_GET[type];                                             //获取信息类型
$state=$_GET[state];                                           //获取信息付费状态
```

```
❶    $sql=mysql_query("update tb_leaguerinfo set checkstate=1 where id=$id");     //更新对应付费信息的状态为已付款
if($sql){                                                                          //如果更新操作成功
❷    echo "<script>alert('该信息已经通过审核！');window.location.href='find_fufei.php?type=$type&state=$state';</script>";    //弹出操作成功提示信息
}
else{                                                                              //如果更新操作失败
     echo "<script>alert('该信息审核操作失败！');history.back();</script>";         //弹出操作失败信息
}
?>
```

代码贴士

❶ update…set：用来修改指定表中的数据。UPDATE 语句的使用方法参见 1.12.2 节。

❷ type=$type&state=$state：将变量 type 与 state 的值重新传到 find_fufei.php 页，目的是为了使 find_fufei.php 页中的$type 和$state 变量重新获得信息类型和付费状态值（管理员选择的检索条件），从而返回到付费信息管理页，并更新数据信息。如果数据处理页不对这两个变量进行传值，那么在返回到付费信息管理页 find_fufei.php 时将会因为检索不到变量的值而出错。

1.12.5 付费信息删除的实现过程

付费信息删除使用的数据表：tb_leaguerinfo

付费信息管理页中“删除”超链接的代码如下：

例程19 代码位置：光盘\TM\01\99pursey\admin\find_fufei.php

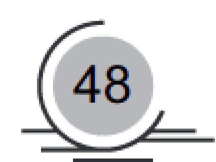
```
<a href="fudel_ok.php?id=<?php echo $info[id];?>&type=<?php echo $type;?>&state=<?php echo $state;?>">删除</a>
```

管理员单击对应主题信息后面的“审核”超链接，将信息所对应的 id 值、信息类型及审核状态传递到数据处理页 fudel_ok.php，用 DELETE 语句将 id 指定的供求信息删除。数据处理页的代码如下：

例程20 代码位置：光盘\TM\01\99pursey\admin\fudel_ok.php

```
<?php
include("../conn/conn.php");                                           //连接数据库文件
$id=$_GET[id];                                                         //获取信息 id 的值
$type=$_GET[type];                                                     //获取信息类型
$state=$_GET[state];                                                   //获取信息付费状态
$sql=mysql_query("delete from tb_leaguerinfo where id=$id");           //删除对应的供求信息
if($sql){                                                              //如果删除操作成功，则弹出提示信息
echo "<script>alert('该信息已经删除！');window.location.href='find_fufei.php?type=$type&state=$state';</script>";
}
else{                                                                  //如果删除操作失败，则弹出提示信息
echo "<script>alert('该信息删除操作失败！');history.back();</script>";
}
?>
```

注意 在删除数据后，仍需要将变量$type 和$state 的值重新传递给付费信息管理页 find_fufei.php，目的是返回到付费信息管理页，查看执行删除操作后的状态。



1.12.6　单元测试

在开发完管理员模块后，对该模块进行单元测试。当管理员对付费供求信息进行审核后，审核操作成功了，但却弹出如图 1.71 所示的错误提示。

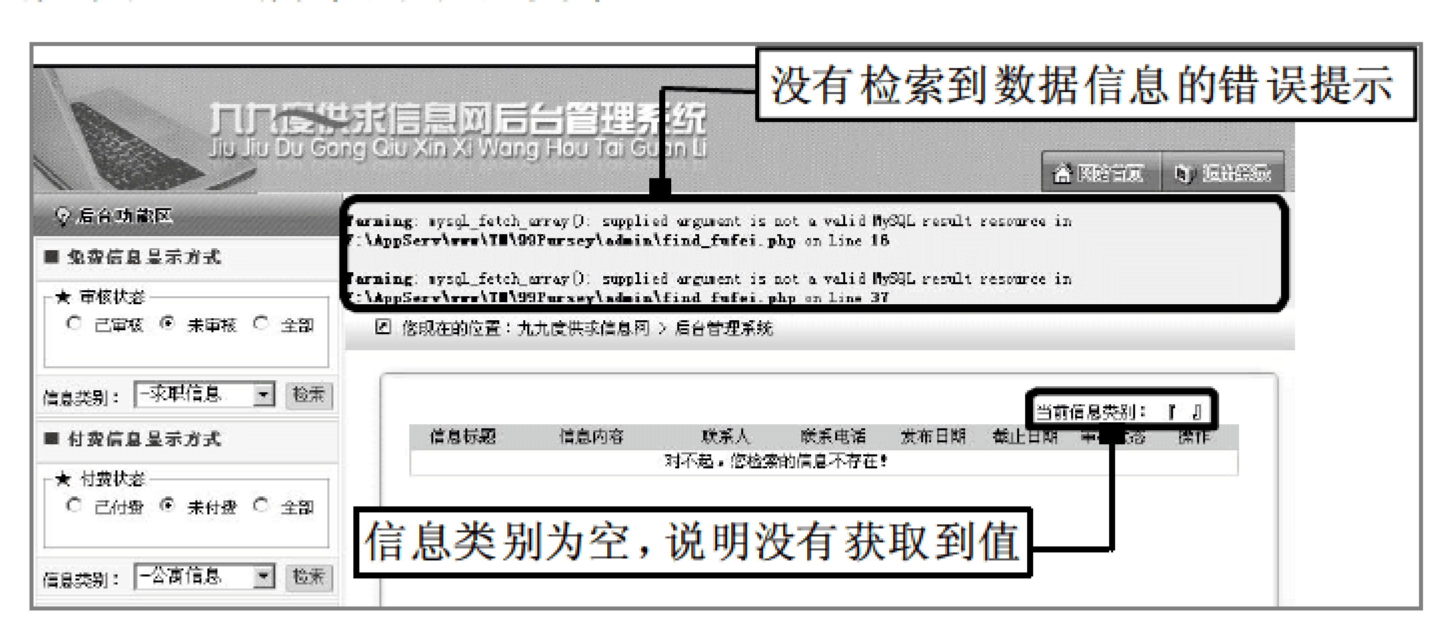


图 1.71　审核付费供求信息的错误提示

在图 1.71 中的错误提示中可以看出，在付费信息管理页的第 16 行和第 17 行出现问题。下面看一下出现问题的这两行代码：

```
if($state=="all"){                              //如果管理员选择的付费状态为“全部”，则执行下面的 SQL 语句
    $sql1=mysql_query("select count(*) as total from tb_leaguerinfo  where type='$type' order by id");
}else{                                          //否则，执行下面的 SQL 语句
    $sql1=mysql_query("select count(*) as total from tb_leaguerinfo  where type='$type' and checkstate=$state order by id");
}
/************************第 16 行代码****************************/
$minfo=mysql_fetch_array($sql1);
/************************第 17 行代码****************************/
$total=$minfo[total];
```

从代码中可以看出，SQL 语句的书写并没有错误。根据图 1.71 所示页面的结果，当前信息类别为空，则说明管理员选择的信息类别没有传过来值。由此可以看出，这是由于在执行审核后页面重新刷新了，因此检索不到管理员选定的查询条件值。

“审核”超链接的源代码如下：

例程21　代码位置：光盘\TM\01\99pursey\admin\find_fufei.php

```
<a href="statefu_ok.php?id=<?php echo $info[id];?>">审核</a>
```

审核处理页的源代码如下：

例程22　代码位置：光盘\TM\01\99pursey\admin\statefu_ok.php

```
<?php
$id=$_GET[id];                                                        //获取信息 id 的值
$sql=mysql_query("update tb_leaguerinfo set checkstate=1 where id=$id");  //更新对应付费信息的状态为已付款
if($sql){                                                             //如果更新操作成功，弹出提示信息
    echo "<script>alert('该信息已经通过审核！');window.location.href='find_fufei.php;</script>";
}
```

解决该问题的方法需要在“审核”超链接传值时将管理员选定的“信息类型”和“审核状态”的变量值一同传递到数据处理页，当审核操作完成后，再将“信息类型”和“审核状态”的变量值重新传递给付费信息管理页 find_fufei.php 即可。

“审核”超链接修改后的代码（加粗的代码部分为修改的代码部分）如下：

例程23　代码位置：光盘\TM\01\99pursey\admin\find_fufei.php

```
<a href="statefu_ok.php?id=<?php echo $info[id];?>&type=<?php echo $type;?>&state=<?php echo $state;?>">
审核</a>
```

技巧　在传递多个变量时，变量之间用“&”符号分隔。

管理员单击对应主题信息后面的“审核”超链接，将信息所对应的 id 值、信息类型及审核状态传递到数据处理页 statefu_ok.php。在执行完更新操作后，需要将“信息类型”和“审核状态”的变量值重新传递到付费信息管理页 find_fufei.php，加粗的代码部分为修改的代码部分。

例程24　代码位置：光盘\TM\01\99pursey\admin\statefu_ok.php

```
<?php
$id=$_GET[id];                                                          //获取信息 id 的值
$type=$_GET[type];                                                      //获取信息类型
$state=$_GET[state];                                                    //获取信息付费状态
$sql=mysql_query("update tb_leaguerinfo set checkstate=1 where id=$id"); //更新对应付费信息的状态为已付款
if($sql){                                                               //如果更新操作成功
    echo "<script>alert('该信息已经通过审核！');window.location.href='find_fufei.php? type=$type&state=
$state';</script>";                                                     //弹出操作成功提示信息
}
```

1.13　网　站　发　布

开发九九度供求信息网的最终目的是将其发布到 Internet 上，供用户浏览访问。

在服务器上上载网站，首先需从服务商处申请固定的 IP 号，然后再注册一个域名，并将域名指定到该 IP 地址。在服务器上安装 PHP 的开发环境，最后上载网站。

下面以九九度供求信息网为例，来讲解在个人服务器上上载网站的过程。

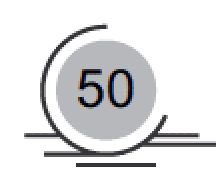
（1）右击“网上邻居”，在弹出的快捷菜单中选择“属性”命令，打开“网络连接”对话框。

（2）右击“本地连接”，在弹出的快捷菜单中选择“属性”命令，打开“本地连接 属性”对话框，如图 1.72 所示。在该对话框的“此连接使用下列选定的组件”列表框中选中“Internet 协议（TCP/IP）”复选框，然后单击“属性”按钮，按服务商提供的指定 IP 设置服务器的 IP 地址，如图 1.73 所示。

（3）安装 PHP 的开发环境，参见本章 1.4 节。

（4）将九九度供求信息网上载到服务器指定的路径下，如图 1.74 所示。

（5）在 IE 浏览器地址栏中输入申请的域名，发布的网站运行结果如图 1.75 所示。



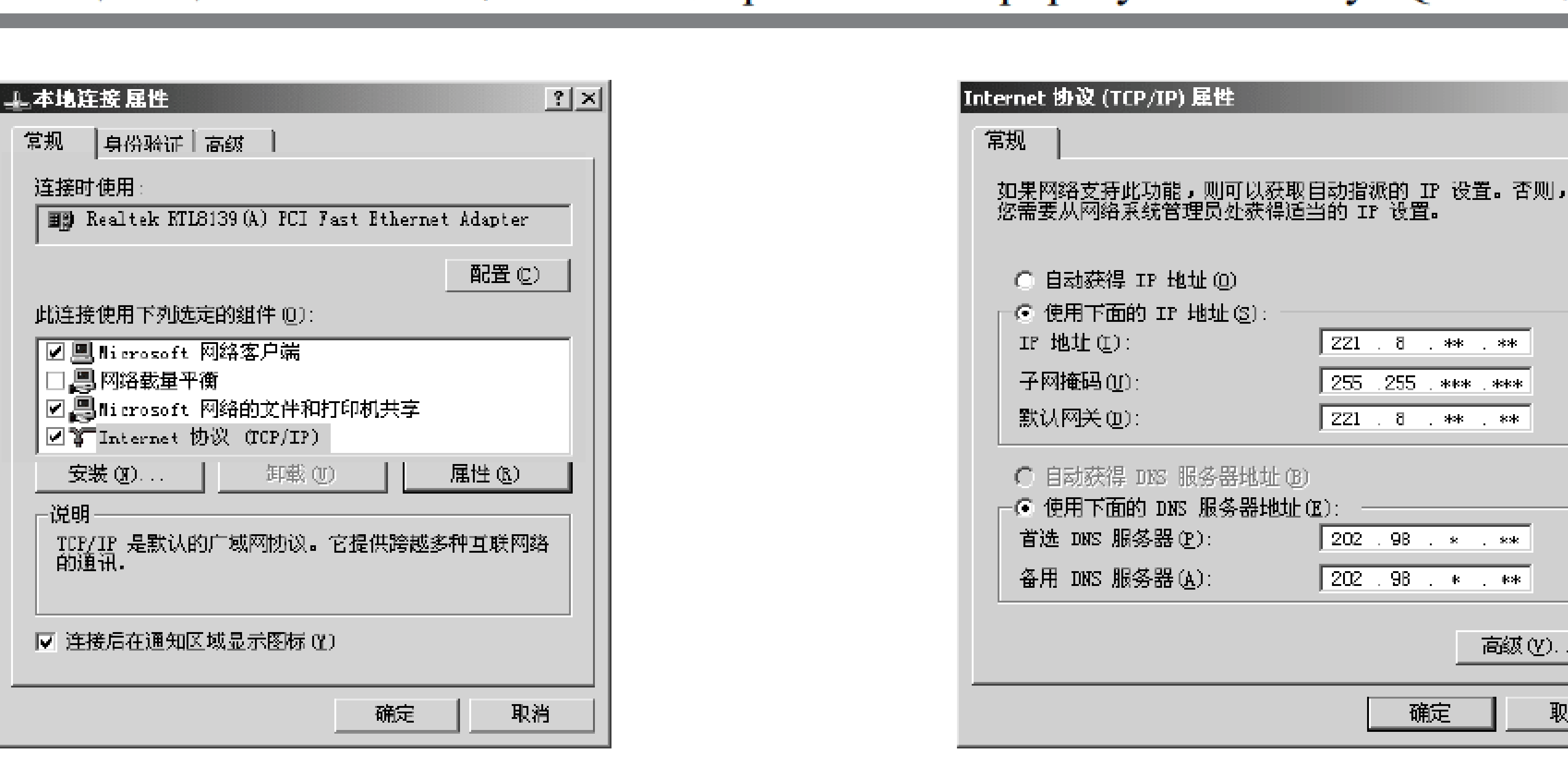


图 1.72　“本地连接 属性”对话框　　　图 1.73　“Internet 协议（TCP/IP）属性”对话框





图 1.74　上载九九度供求信息网到指定的路径下　　　图 1.75　九九度供求信息网发布后的运行结果

1.14　开发技巧与难点分析

1.14.1　查询关键字描红功能

在九九度供求信息网前台信息检索过程中体现了方便快捷的人性化原则，为了方便浏览者查阅信息，便于查找与浏览者的关键字相符合的信息，在搜索引擎中添加描红功能。

查询关键字描红是指将查询关键字以特殊的颜色、字号或字体进行标识，这样可以使浏览者快速找到所需的关键字，方便浏览者从搜索结果中查找所需内容。查询关键字描红适用于模糊查询。下面介绍如何实现查询关键字描红。

本系统用 str_ireplace()函数来替换查询关键字，当显示所查询的相关信息时，将输出的关键字的字体替换为红色。描红功能的业务流程如图 1.76 所示。

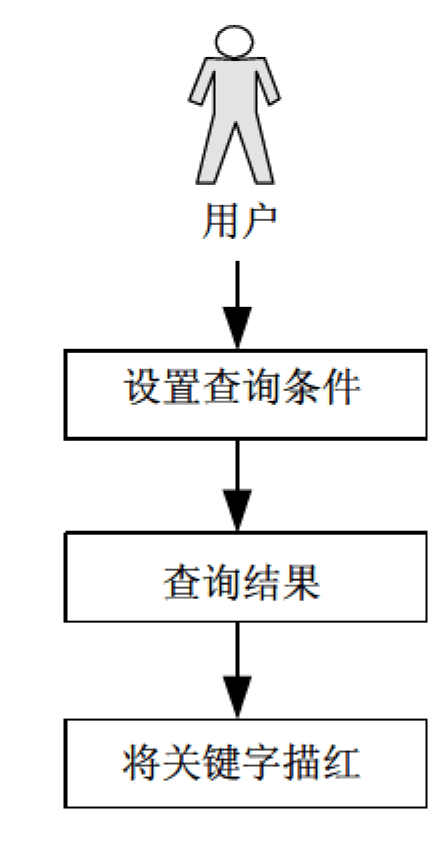
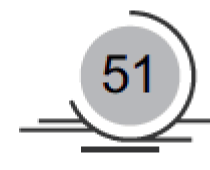


图 1.76　描红功能的业务流程

str_ireplace()函数用于将某个指定的字符串替换为另一个指定的字符串，不区分大小写。该函数的语法如下：



```
mixed str_ireplace ( mixed search, mixed replace, mixed subject [, int &count])
```

该函数将所有在参数 subject 中出现的参数 search 以参数 replace 取代，参数&count 表示取代字符串执行的次数。

str_ireplace()函数的参数说明如表 1.12 所示。

表 1.12　str_ireplace()函数的参数说明

参　　数	说　　明
search	必要参数，指定需要查找的字符串
replace	必要参数，指定替换的值
subject	必要参数，指定查找的范围
count	可选参数，获取执行替换的数量

注意 该函数在执行替换的操作时，是不区分大小写的。如果需要对大小写加以区分，可以使用 str_replace()函数。

本系统应用 str_ireplace()函数替换查询字符串为红色的字符串，关键代码如下：

```
<?php
include("conn/conn.php");                                              //连接数据库文件
$content=$_POST[content];                                              //获取查询关键字
$sql1=mysql_query("select * from tb_leaguerinfo  where checkstate=1 and type='$type' and content
like'%$content%' or title like'%$content%' or linkman like'%$content%' or tel like'%$content%'");
$info1=mysql_fetch_array($sql1);                                       //采用模糊信息资源查询
//下面应用 str_ireplace()函数将指定的查询关键字用红色文字替代，并输出替换后的字符串
echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[type]);//替换信息类型为红色字体
echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[title]); //替换信息标题为红色字体
echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[edate]);  //替换发布时间为红色字体
echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[content]); //替换信息内容为红色字体
echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[linkman]); //替换联系人为红色字体
echo str_ireplace($content,"<font color='#FF0000'>".$content."</font>",$info1[tel]);  //替换联系电话为红色字体
?>
```

1.14.2　表单数据的提交方式

获取表单元素提交的值是表单应用中最基本的操作方法。表单数据的传送方法有 get 方法和 post 方法两种，通过<form>的 method 属性来指定。下面来具体讲解一下这两种方法在实际工作中的应用范围和使用技巧。

1．通过 get 方法提交数据

使用 get 方法时，表单数据被当作 url 的一部分一起传过去。格式如下：

```
http://url?name1=value1&name2=value2…
```

☑　url：表单响应地址。例如，127.0.0.1/index.php。



☑ name：表单元素的名称。例如，<input type="text" name="user">，这里 name 的属性值就是 user。通过 name 值可以获取 value 的属性值。
☑ value：表单元素的值。例如，<input type="text" name="user" value="mr">，意思是名字叫 user 的 text 表单元素的值为 mr。

说明 url 和表单元素之间用“?”隔开，而多个表单元素之间用“&”隔开，每个表单元素的格式都是“name=value”，固定不变。

PHP 使用$_GET 预定义变量自动保存通过 get 方法传过来的值，使用格式为：

```
$_GET[name]
```

这样，就可以直接使用名字为 name 的表单元素的值。

技巧 有的 PHP 版本中直接写$name 就可以调用表单元素的值，这和 php.ini 文件的配置有关系，定位到 GLOBAL=ON/OFF 行，如果值为 ON，就可以直接写成$name，反之则不可以。直接应用表单名称十分方便，但也存在一定的安全隐患。推荐读者关闭 GLOBAL 项。

使用文本框传值的程序中包含一个文本框元素。在文本框中输入信息，当单击“提交”按钮时，文本框内的信息就会和 url 一起显示在地址栏中。代码如下：

```
<form name="login" method="get" action="index.php">
…
</form>
```

技巧 get 方法是使用 url 来进行传值的，“加入收藏夹”的功能就是 get 方法的应用之一。想要将一个带参数的网址加入到收藏夹中，只能使用 url，它可以脱离表单的束缚。文字、图片等都可以使用这种方法来传值。

2．通过 post 方法提交数据

get 方法有个最大的缺点，就是它的信息是显示在客户端浏览器上的，这使用户的资料暴露无遗，而且 url 本身受长度限制（1024KB），不能传输较大的数据。这时可以选择 post 方法。使用时，将<form>表单中的属性 method 设置成 post 即可。post 方法不依赖 url，所有提交的信息在后台传输，不会显示在地址栏中，安全性高，而且没有长度限制。

使用 PHP 的$_POST[name]变量可以获取表单元素的值，格式和$_GET[name]类似：

```
$_POST[name]
```

例如，使用的 post 方法返回文本框信息，代码如下：

```
<form name="login" method="post" action="index.php">
…
</form>
```

1.15　MySQL 数据库技术专题

phpMyAdmin 是众多 MySQL 图形化管理工具中应用最广泛的一种，它是基于 PHP 语言编写的，该工具是 B/S 结构、基于 Web 跨平台的管理程序，并且支持简体中文，可以在官方网站免费下载。安装后在浏览器地址栏中输入 http://127.0.0.1/phpMyAdmin/，即可进入 MySQL 的管理界面。

phpMyAdmin 为 Web 开发人员提供了类似于 Access、SQL Server 的图形化数据库操作界面，通过该管理工具可以进行绝大部分的 MySQL 操作，包括对数据库及数据表的建立和维护。

1.15.1　创建和删除数据库

1．创建数据库

在 phpMyAdmin 的主界面中有两个文本框和“创建”按钮，首先在文本框中输入数据库的名称，然后选择编码，最后单击“创建”按钮，这样新的数据库就可以被创建成功。例如，创建一个名称为 db_pursey 的数据库，首先在文本框中输入 db_pursey，之后在下拉列表框中选择要使用的编码，在 Windows 下一般选择 gb2312_chinese_ci，如图 1.77 所示。最后单击“创建”按钮，这样名为 db_pursey 的数据库就被创建成功，执行结果如图 1.78 所示。

该数据库名称出现在左侧导航栏的数据库下拉菜单中，选择这个数据库，在右侧界面中可以对该数据库进行操作，如结构、SQL、导出、搜索、查询、删除等，单击相应的按钮即可进入相应的操作界面。但是在创建数据库还没有创建数据表的情况下，只能够执行结构、SQL 和删除 3 项操作，其他操作不可以执行。



图 1.77　phpMyAdmin 管理界面

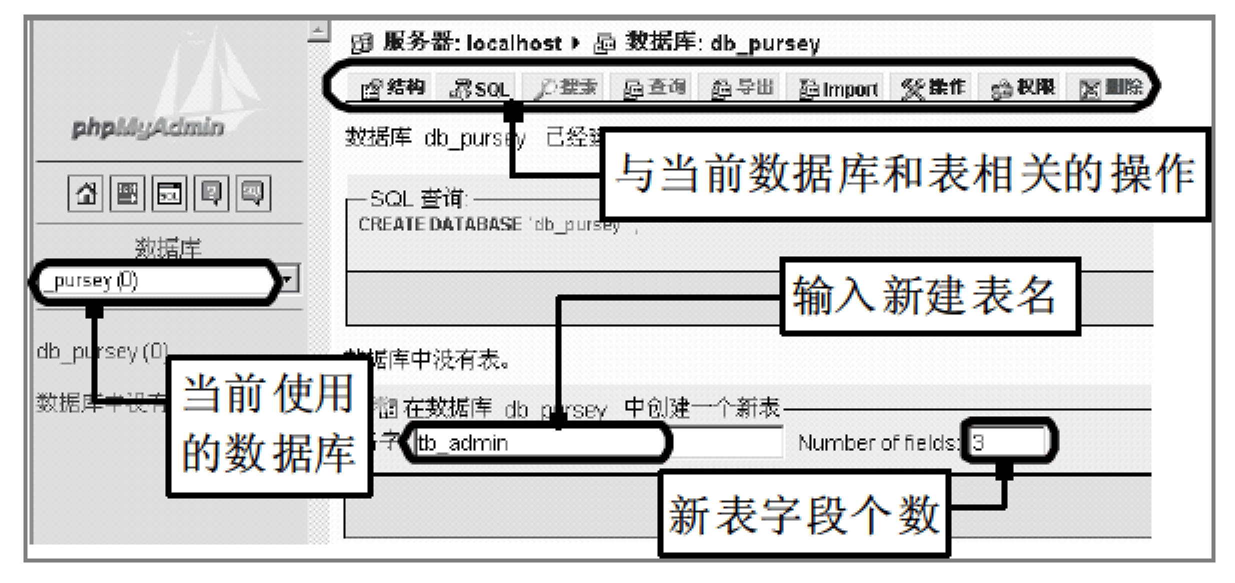


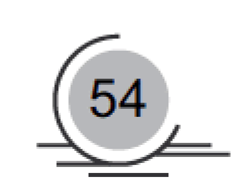
图 1.78　数据库的建立

2．删除数据库

要删除某个数据库，首先在左侧的下拉菜单中选择该数据库，然后单击右侧界面中的“删除”按钮即可。

1.15.2　创建和删除数据表

针对表级操作是在选定了数据库的情况下进行的，即表级操作的前提是用户必须选择一个数据库，在该数据库中进行表的建立和维护。





1．创建数据表

创建数据库 db_pursey 后，在右侧页面中会出现如图 1.78 所示的数据表创建提示页面，完成数据表的创建操作。

首先在表单中输入数据表的名称和字段数，然后单击“执行”按钮，进入各个字段的详细信息录入表单，包括字段名、数据类型、长度/值、属性、默认值、额外和索引的类型等，在这里就完成了对表结构的详细设置，如图 1.79 所示。

当所有的信息都输入完以后，就可以单击“保存”按钮，成功创建数据表 tb_admin。一个新的数据表被创建后，进入到数据表页面，在这里可以通过改变表的结构来修改表，可以执行添加新的列、删除列、索引列、修改列的数据类型或者字段的长度/值等操作，如图 1.80 所示。



图 1.79　创建表信息

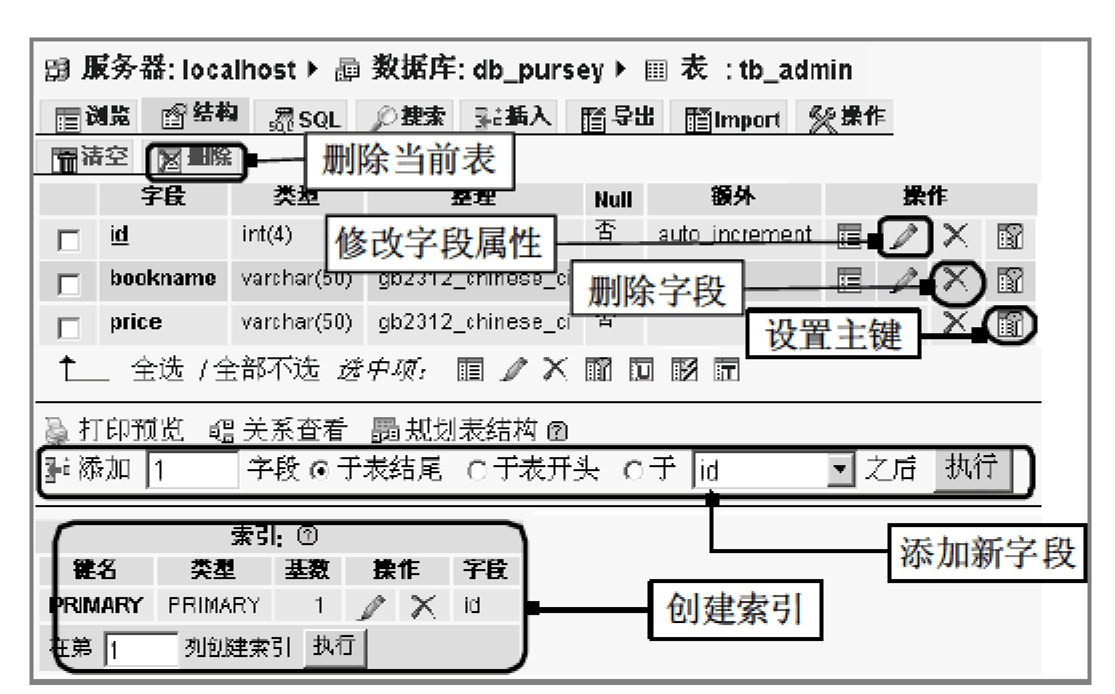


图 1.80　操作列表

2．删除数据表

执行删除表的操作很简单，只要单击图 1.80 所示页面上方的“删除”按钮，就可以轻松地删除当前数据表。

1.16　本 章 总 结

本章依据软件开发流程介绍了九九度供求信息网的开发过程。在开发任何一个项目前，首先要充分做好前期准备，如完善的需求分析、清晰的业务流程、合理的程序结构等，这样在后期的程序开发中才会得心应手，有备无患。通过本章的学习，读者可以了解数据库建模的概念，掌握 PowerDesigner 数据库建模的方法，熟悉框架技术在 Web 应用程序中的应用。

第 2 章

BCTY365 网上社区

（Apache+PHP+phpMyAdmin+MySQL 5.0 实现）

随着人类文明的不断进步，网络这个虚拟世界也在发生着变化，现实世界中的所有内容在网络上都有反映。现实世界中物以类聚，人以群分，网络世界中也形成了各式各样的社区。这种社区建立在特殊的兴趣点（如体育、新闻、宠物、电影和游戏等）、特殊的人群、特殊的爱好或者是特殊的服务上等。社区也是一种特定的商业模式，它本身可以进行电子商务，也可以完全与商务无关。

所谓网上社区是指包括BBS/论坛、聊天室、博客等形式在内的网上交流空间，同一主题的网上社区集中了具有共同兴趣的访问者，由于有众多用户的参与，因此具备了交流的功能，成为一个营销场所。

网上社区有各种不同的表现形式和规模，有个人创办的社区，功能和界面追求时尚、个性突出；有大型的商业性质社区，以盈利为目的，分类多元化，适合不同类型的网民。

本章开发的BCTY365网上社区主要面向程序开发人员，集论坛、留言板、软件下载、升级下载、技术支持和在线购物等功能于一身，既是一个程序开发者交流的平台，更是一个网络营销的场所。通过阅读本章，可以学习到：

- 网上社区开发的基本过程
- 如何做需求分析和系统设计
- 在 Linux 操作系统下搭建 PHP 开发环境
- 如何设计和创建数据库、数据表
- 如何设计公共类
- 软件上传和下载功能的实现方法
- 在线论坛功能的实现方法
- 在 Linux 操作系统下发布网站
- 在线支付技术

2.1 开发背景

随着市场竞争的日益激烈，企业的生存和发展之路更加艰难，要想使企业保持旺盛的生命力，企业必须要跟上时代发展的脚步，不断为企业注入新的活力。某科技公司为适应市场的需求，增加公司在互联网上的影响力，将开发一个网上社区系统，为广大的编程爱好者提供一个交流的平台，并且以此来推广该公司的软件产品。

2.2 系统分析

当一个开发项目被确立时，首先要做的就是需求分析、可行性分析，然后编写项目计划书，以使项目开发人员了解和掌握网站的前期策划和网站开发流程。

2.2.1 需求分析

在开发网上社区之前，首先要明确所要开发的社区属于什么类型，是个人的社区系统，还是商业化的社区系统；并且要知道开发的社区是面向什么样的人群，是普通网民，还是专业的技术人员，或者是其他的特殊群体。针对不同的人群，社区应该具有不同的特点。当明确了这些，项目开发的思路就清晰了，然后再对网站上一些相关的社区进行考察、分析，从中吸取经验，并结合企业的要求以及实际的市场调查结果，提出一个合理的网上社区网站功能架构。本网站需求如下：

- ☑ 网站设计页面要求整洁、美观大方，能够展示企业形象。
- ☑ 网站页面具有 Banner 广告，树立企业良好的口碑宣传。
- ☑ 设计主要从编程者的角度考虑，为编程者解决在开发中出现的问题。
- ☑ 展示出企业全力推出的软件产品和提供的免费软件，以此吸引浏览者。
- ☑ 提供一个良好的网上购物的操作平台。
- ☑ 提供技术支持，解决编程过程中常见的问题。
- ☑ 提供一个讨论和研究问题的平台。
- ☑ 做到让广大浏览者关注企业的动态。
- ☑ 为客户提供反馈信息的平台，能够做到及时与客户进行沟通。
- ☑ 完善的后台管理系统。

2.2.2 可行性分析

可行性分析是世界上普遍采用的一种研究工程项目是否可行的科学。其通过各种有效的方法，对工程项目进行分析，从技术、经济、市场等方面加以评价，最终给投资决策者提供是否选择该项目进行开发的依据。

BCTY365 网上社区项目开发的可行性分析主要从以下两个方面考虑。

1．经济可行性分析

企业为扩大公司的影响力，推出软件产品，采用网上社区的形式在网络上进行推广，不但可以汇聚更多的人气，而且可以让更多的人了解该企业，从而达到推广企业软件产品的目的，最终为企业带来更大的收益。更重要的一点是采取该方法的成本相对其他的电视广告或者人力宣传的成本要低得多，虽然周期很长，但是却能够取得长期的收益。

2．技术可行性分析

网上社区系统的开发采用的是 Apache+PHP+phpMyAdmin+MySQL 5.0，开发软件都是免费的，可以直接从网上下载，无须支付任何费用。要完成 BCTY365 网上社区系统的开发，必须能够配置 PHP 程序开发的环境，掌握在线支付、购物车和在线论坛技术。

2.2.3 编写项目计划书

根据《GB8567－88 计算机软件产品开发文件编制指南》中的项目开发计划要求，结合单位实际情况，设计项目计划书如下。

1．引言

（1）编写目的

为了保证项目开发人员按时保质地完成预订目标，更好地了解项目实际情况，按照合理的顺序开展工作，现以书面的形式将项目开发生命周期中的项目任务范围、项目团队组织结构、团队成员的工作责任、团队内外沟通协作方式、开发进度、检查项目工作等内容描述出来，作为项目相关人员之间的共识和约定，作为项目生命周期内的所有项目活动的行动基础。

（2）背景

BCTY365 网上社区系统是本公司与×××信息科技有限公司签定的待开发项目，网站性质为信息服务类型，为企业与客户、浏览者和会员之间提供一个技术交流平台，并且全力推出企业的软件产品。项目周期为两个月，项目背景规划如表 2.1 所示。

表 2.1　项目背景规划

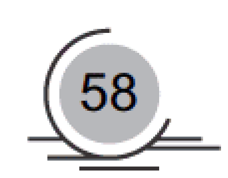
项 目 名 称	签定项目单位	项目负责人	项目承担部门
BCTY365 网上社区系统	甲方：×××信息科技有限公司	甲方：赵经理	设计部门 开发部门 测试部门
	乙方：×××网络科技有限公司	乙方：张经理	

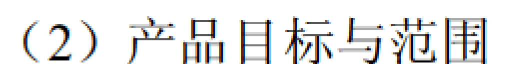
2．概述

（1）项目目标

项目目标应当符合 SMART 原则，把项目要完成的工作用清晰的语言描述出来。BCTY365 网上社区系统的项目目标如下：

BCTY365 网上社区系统主要用于在网络中树立企业的形象，为程序开发者提供一个交流的平台，拉近企业与客户、会员和浏览者之间的距离，从而达到推广企业软件产品的目的。整个项目需要两个月结束，交给客户进行验收。





（2）产品目标与范围

一方面 BCTY365 网上社区系统能够为企业节省大量人力资源，企业不再需要大量的业务人员去跑市场，间接为企业节约了成本。另一方面，BCTY365 网上社区系统能够收集海量编程问题的解决方案和好的建议，将会有大量用户访问网站，有助于提高企业知名度。

（3）应交付成果

项目开发完成后，交付的内容如下：

☑ 以光盘的形式交付 BCTY365 网上社区系统的源程序、网站数据库文件、系统使用说明书。

☑ 客户方应用自己的服务器，因此需要乙方架设 Apache 服务器，安装 PHP 开发环境，协助甲方购买域名，将开发的 BCTY365 网上社区系统发布到互联网上运行。

☑ 网站发布到互联网上以后，进行后期的 6 个月无偿维护与服务，超过 6 个月后进行网站有偿维护与服务。

（4）项目验收方式与依据

项目验收分为内部验收和外部验收两种方式。在项目开发完成后，首先进行内部验收，由系统测试员根据用户需求和项目目标进行验收。项目在通过内部验收后，交给用户进行验收，验收的主要依据为需求规格说明书。

3．项目团队组织

（1）组织结构

为了完成 BCTY365 网上社区系统的项目开发，公司组建一个临时的项目团队，由项目经理、系统分析师、PHP 开发工程师、网页设计师和系统测试员构成，如图 2.1 所示。

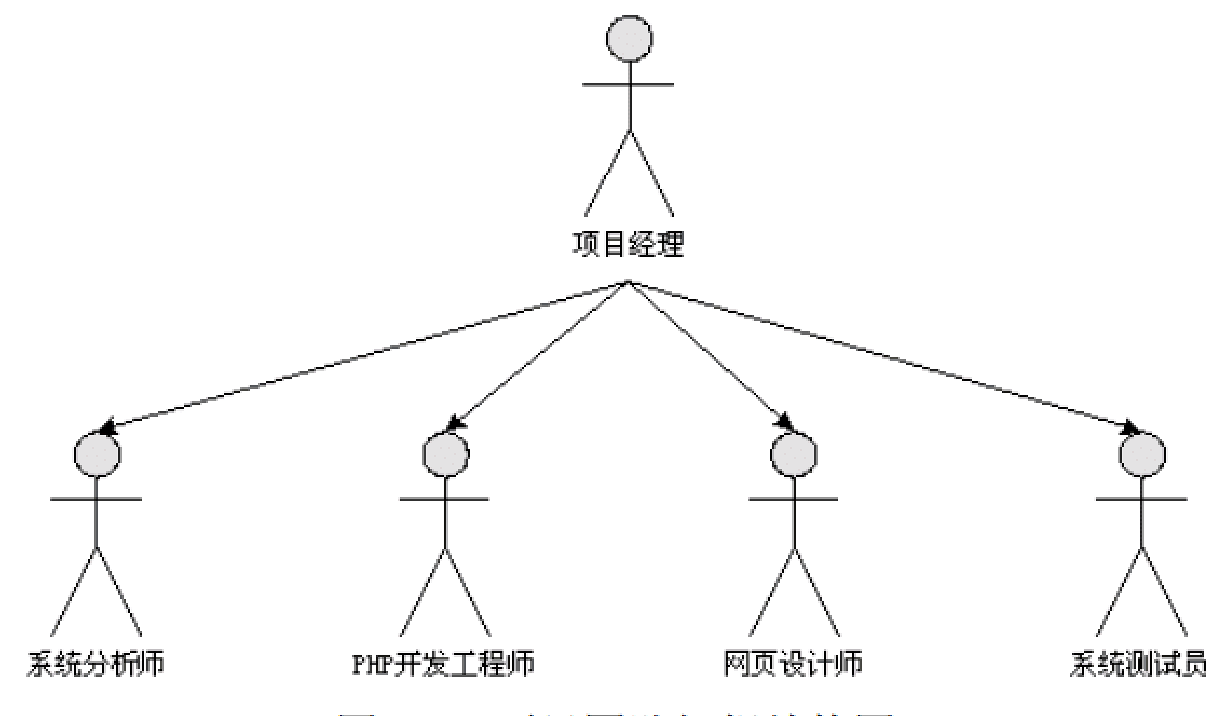


图 2.1　项目团队组织结构图

（2）人员分工

为了明确项目团队中每个人的任务分工，现制定人员分工表，如表 2.2 所示。

表 2.2　人员分工表

姓　名	技术水平	所属部门	角　色	工作描述
张达明	MBA	项目开发部	项目经理	负责项目的审批、决策的实施、项目的前期分析、策划、项目开发进度的跟踪、项目质量的检查
王言辉	高级系统分析师	项目开发部	系统分析师	负责系统功能分析、系统框架设计
潘攀	高级 PHP 工程师	项目开发部	PHP 开发工程师	负责软件前后台设计与编码
刘悦	高级美工设计师	设计部	网页设计师	负责网页风格的确定、网页图片的设计
齐欣	高级系统测试工程师	项目开发部	系统测试员	对软件进行测试、编写软件测试文档

2.3 系统设计

2.3.1 系统目标

根据对目前网络上各种社区的分析和研究，结合本项目的实际需求，在设计时应该满足以下目标：

☑ 界面设计美观大方、方便、快捷、操作灵活，树立企业形象。
☑ 功能完善、结构清晰。
☑ 重点突出企业的软件产品。
☑ 及时更新网站公告。
☑ 及时查阅和回复客户反馈信息。
☑ 为用户提供沟通和交流的平台。
☑ 购物车模块的设计结构合理、流程清晰。
☑ 购物结算功能设计符合逻辑，计算准确。
☑ 订单处理功能的设计及时、准确、安全。
☑ 处理好网上支付功能的设计与网上银行之间数据的传递。
☑ 具备完善的后台管理功能，能够及时、准确地对网站进行维护和更新。
☑ 系统运行稳定，具备良好的防范措施。

2.3.2 系统功能结构

结合需求分析和系统目标中的内容，BCTY365 网上社区系统的功能结构已经设计完成。为了使读者能够更清楚地了解网站的结构，下面给出 BCTY365 网上社区前台和后台功能模块结构图。

BCTY365 网上社区前台管理系统的功能设计如图 2.2 所示。

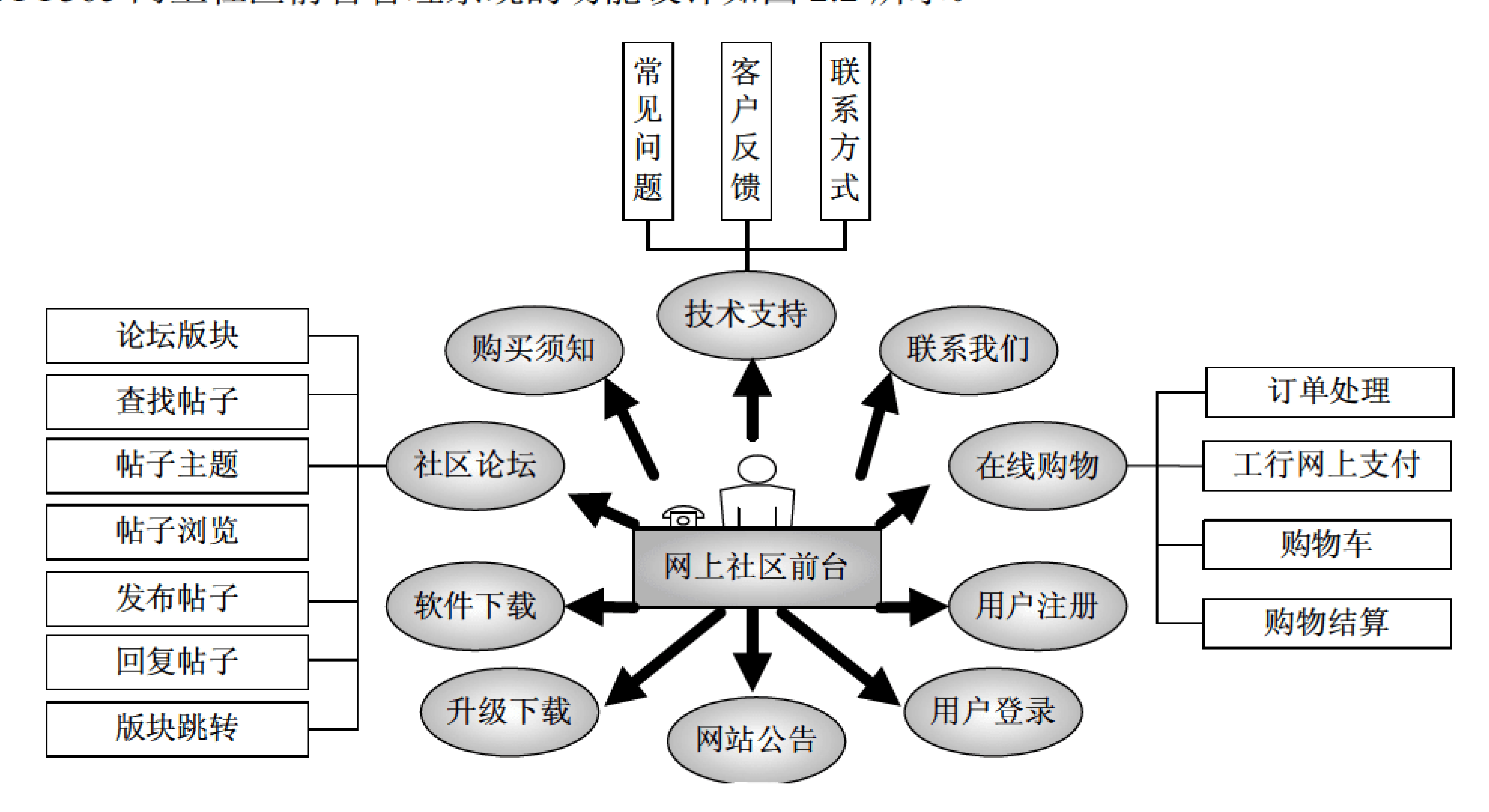


图 2.2　网上社区前台功能模块结构图

BCTY365 网上社区后台管理系统的功能设计如图 2.3 所示。

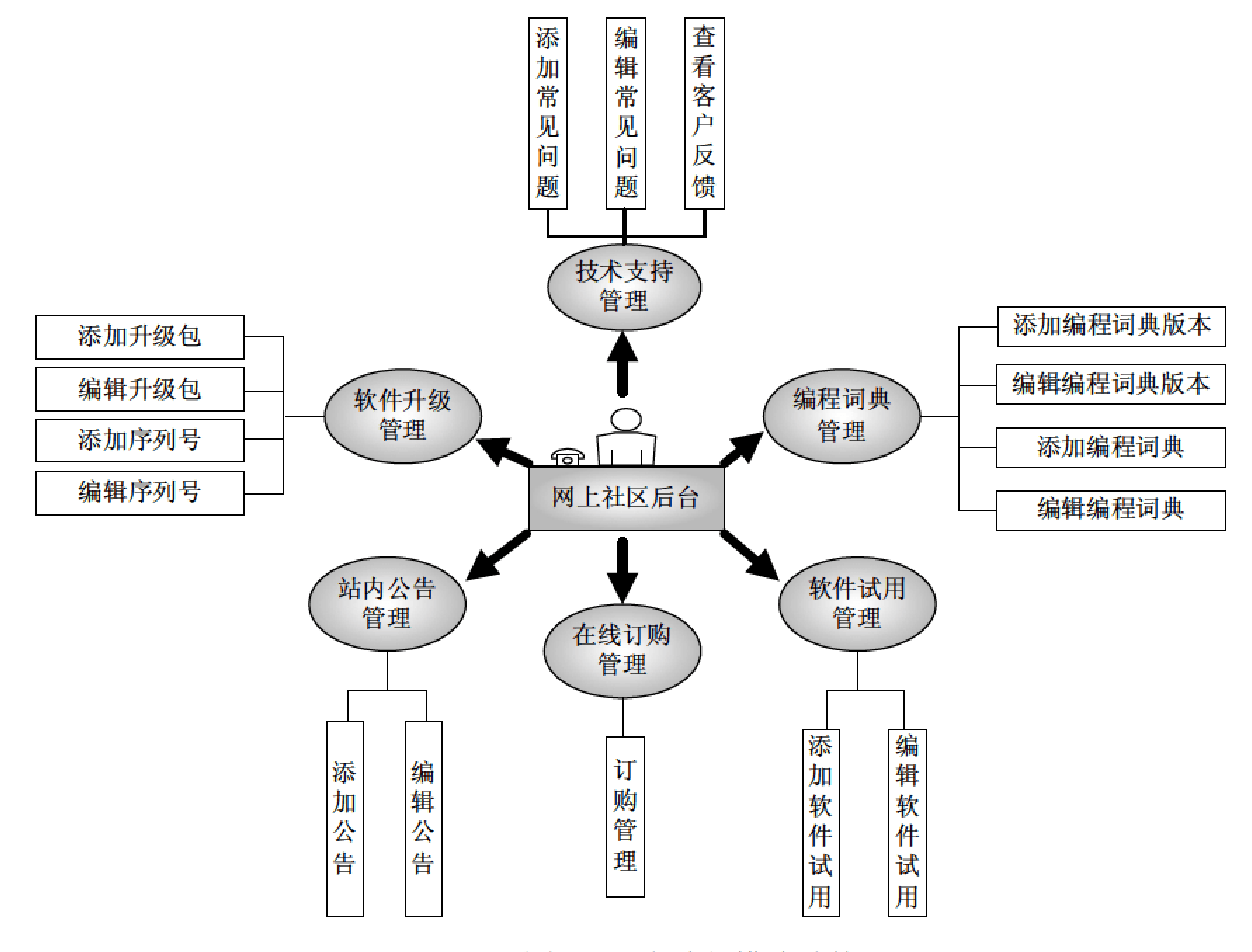


图 2.3　网上社区后台功能模块结构图

2.3.3　系统预览

BCTY365 网上社区系统由多个程序页面组成，下面给出几个典型页面，其他页面参见光盘中的源程序。

前台首页如图 2.4 所示，该页面用于展示本系统的功能模块，突出企业的形象，推广企业的软件产品。后台首页如图 2.5 所示，该页面用于实现对编程词典、技术支持、软件升级、软件试用等内容的管理。



图 2.4　前台首页（光盘\TM\02\bcty365\index.php）



图 2.5　后台首页（光盘\TM\02\bcty365\ admin\default.php）

在线订购模块的页面效果如图 2.6 所示，该页面主要用于展示本企业在线推出的软件产品，实现对

产品的在线购买功能。软件下载模块的页面效果如图 2.7 所示，该页面主要用于展示本企业提供的免费软件，并且提供下载链接。



图 2.6　在线订购（光盘\TM\02\bcty365\ morebccd.php）　　图 2.7　软件下载（光盘\TM\02\bcty365\ rjxz.php）

社区论坛模块的页面效果如图 2.8 所示，该页面主要用于展示论坛中的各大版块，并且提供超链接跳转到对应的版块。后台的登录页面效果如图 2.9 所示，该页面主要实现后台管理员登录。



图 2.8　社区论坛（光盘\TM\02\bcty365\ bbs_index.php）



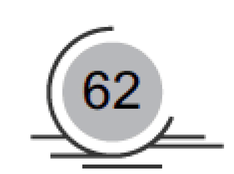
图 2.9　后台登录（光盘\TM\02\ bcty365\admin\index.php）

2.3.4　开发环境

在开发 BCTY365 网上社区时，该项目使用的软件开发环境如下。

1．服务器端

☑ 操作系统：Windows Server 2003/Linux（推荐）。
☑ 服务器：Apache 2.2.8。
☑ PHP 软件：PHP 5.2.6。
☑ 数据库：MySQL 5.0.51。
☑ MySQL 图形化管理软件：phpMyAdmin-2.10.3。
☑ 开发工具：Dreamweaver 8。
☑ 浏览器：IE 6.0 及以上版本。
☑ 分辨率：最佳效果为 1024×768 像素。





2. 客户端

- ☑ 浏览器：IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

2.3.5　文件夹组织结构

在进行网站开发之前，要对网站的整体文件夹组织架构进行规划，对网站中使用的文件进行合理的分类，分别放置于不同的文件夹下。通过对文件夹组织架构的规划，可以确保网站文件目录明确、条理清晰，同样也便于网站后期的更新和维护。本案例的文件夹组织结构如图 2.10 所示。

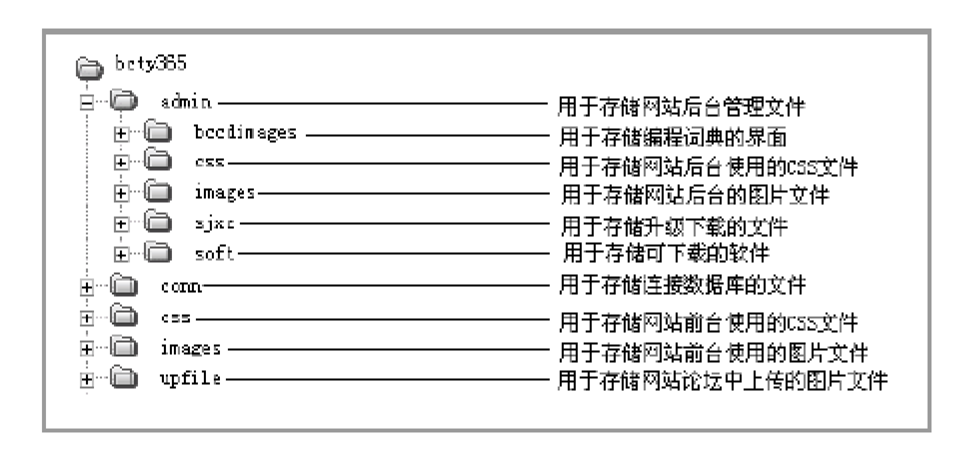


图 2.10　文件夹组织结构

2.4　在 Linux 操作系统下搭建 PHP 开发环境

Red Hat Linux 9 是 Linux 众多版本中比较大众化的一版。在安装系统时，如果选择完全安装或者选择 Apache、MySQL、PHP 的安装包，则三者将被安装到系统中，用户只需将 Apache 和 MySQL 服务启动就可以使用二者，非常方便，但是 Apache 和 MySQL 的版本可能不是很理想。为了能够创建一个良好的 PHP 开发环境，这里将详细介绍自行在 Linux 下安装和配置 Apache 2+MySQL 5.0+PHP 5 的方法。

首先应该到相关官方网站下载三者的安装包和 libxml2 的安装包：

- ☑ httpd-2.0.58.tar.gz 或更高版本（http://httpd.apache.org/）。
- ☑ mysql-standard-5.0.0-alpha-pc-linux-i686.tar.gz 或更高版本（http://www.mysql.com/）。
- ☑ php-5.0.0.tar.gz 或更高版本（http://www.php.net/）。
- ☑ libxml2-2.5.10.tar.gz 或更高版本（如果读者系统中 libxml2 的版本已经等于或高于该版本，可以不必下载该安装包）。

2.4.1　Linux 下 Apache 的安装配置

首先将下载的 httpd 安装包复制到适当的位置，例如/usr/local/work 下（如果目录不存在，可以建立该目录）。打开 Red Hat Linux 9 的主菜单，选择“系统工具”，在弹出的菜单中选择“终端”命令，将打开如图 2.11 所示的终端对话框。

说明　Linux 下 Apache、MySQL 及 PHP 的安装都是在如图 2.11 所示的终端命令窗口中通过命令方式实现的。



图 2.11　Red Hat Linux 9 的终端命令对话框

在该对话框中输入如下命令进入 work 目录：

```
cd /usr/local/work
```

在 work 目录中输入如下命令解压 httpd-2.0.58.tar.gz：

```
tar xfz httpd-2.0.58.tar.gz
```

进入解压后的目录 httpd-2.0.58：

```
cd httpd-2.0.58
```

建立 makefile，并将 Apache 2 安装到/usr/local/apache2 目录下：

```
./configure –prefix=/usr/local/apache –enable-module=so
```

开始编译：

```
make
```

开始安装到设置的目录中：

```
make install
```

到此 Apache 2 的安装工作完成，可以在每次启动系统时通过如下命令启动或重新启动 Apache 2 服务：

```
/usr/local/apache2/bin/apachectl start
/usr/local/apache2/bin/apachectl restart
```

打开浏览器，在地址栏中输入 http://127.0.0.1 或者 http://localhost，按 Enter 键，如果出现如图 2.12 所示的页面，则说明 Apache 2 安装成功。

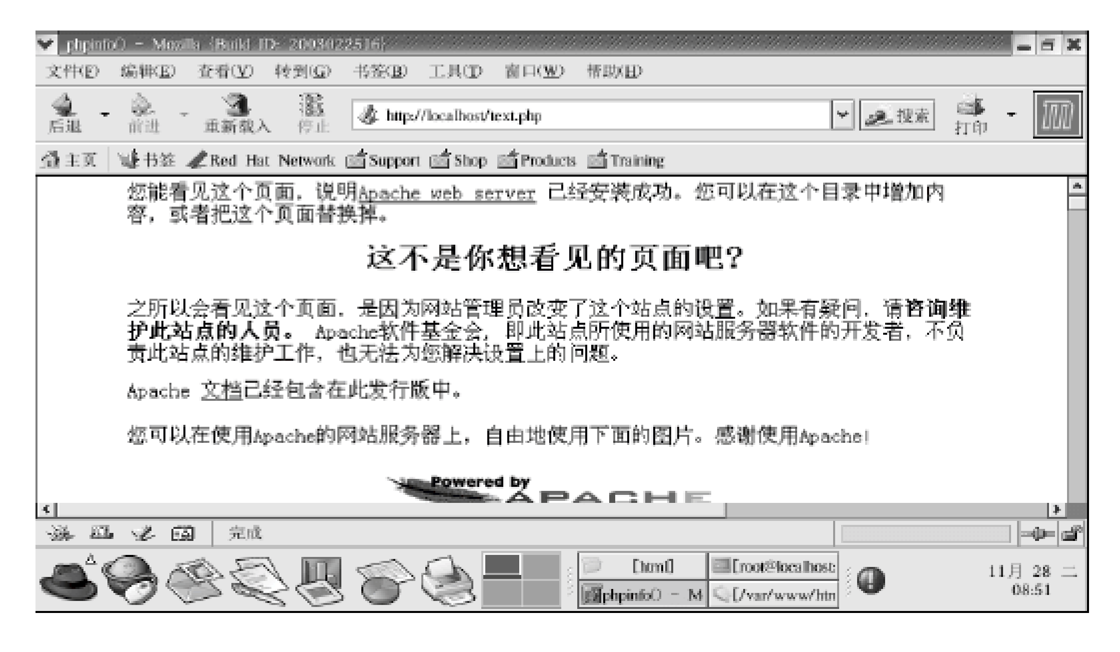
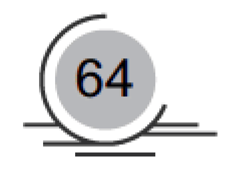


图 2.12　测试 Apache 服务器

2.4.2　Linux 下 MySQL 的安装配置

将 mysql-standard-5.0.0-alpha-pc-linux-i686.tar.gz 复制到/usr/local/work 目录下，建立 MySQL 账号：

```
groupadd mysql
```



在组群中加入 MySQL：

```
useradd –g mysql mysql
```

进入 local 目录：

```
cd /usr/local
```

将 mysql-standard-5.0.0-alpha-pc-linux-i686.tar.gz 解压到该目录：

```
tar xfz /usr/local/work/mysql-standard-5.0.0-alpha-pc-linux-i686.tar.gz
```

考虑到 MySQL 数据库升级的需要，所以通常以链接的方式建立/usr/local/mysql 目录：

```
ln –s mysql-standard-5.0.0-alpha-pc-linux-i686.tar.gz mysql
```

进入 MySQL 目录：

```
cd mysql
```

在/usr/local/mysql/data 中建立 MySQL 的数据库：

```
scripts/mysql_install_db –user=mysql
```

修改文件权限：

```
chown –R root .
chown –R mysql data
chgrp –R mysql .
```

到此 MySQL 安装成功。可以通过在终端中输入如下命令启动 MySQL 服务：

```
/usr/local/mysql/bin/mysqld_safe –user=mysql &
```

启动 MySQL 后输入如下命令查看安装结果：

```
/usr/local/mysql/bin/mysql –uroot
```

如果终端对话框出现如图 2.13 所示的提示，则说明 MySQL 安装成功。

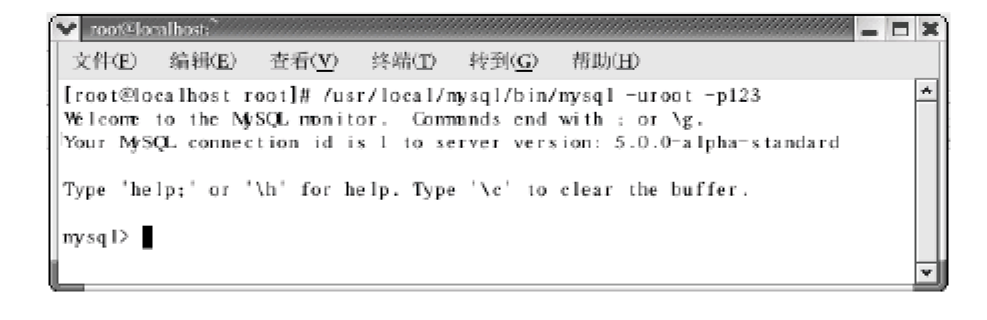


图 2.13　测试 MySQL

2.4.3　Linux 下 PHP 的安装配置

首先查看系统中 libxml2 的版本号，如果 libxml2 的版本号小于 2.5.10，则需要安装 libxml2-2.5.10.tar.gz 或更高版本，因为 PHP 5 必须在 libxml2 的版本大于 2.5.10 的前提下才能够安装。

将 libxml2-2.5.10.tar.gz 复制到/usr/local/work 目录下，并进入该目录：

```
cd /usr/local/work
```

解压 libxml2-2.5.10.tar.gz：

```
tar xfz libxml2-2.5.10.tar.gz
```

进入该目录：

```
cd libxml2-2.10
```

建立 makefie 并将 libxml2 安装到/usr/local/libxml2 下：

```
./configure –prefix=/usr/local/libxml2
```

开始编译：

```
make
```

开始安装到设置的目录去：

```
make install
```

到此 libxml2 安装成功。
将 php-5.0.0.tar.gz 复制到/usr/local/work 目录下，并进入该目录：

```
cd /usr/local/work
```

解压 php-5.0.0.tar.gz：

```
tar xfz php-5.0.0.tar.gz
```

进入 php-5.0.0 目录：

```
cd php-5.0.0
```

建立 makefile：

```
./configure –with-apxs2=/usr/local/apache2/bin/apxs \
--with-mysql=/usr/local/mysql \
--with-libxml-dir=/usr/local/libxml2
```

开始编译：

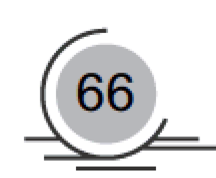
```
make
```

开始安装：

```
make install
```

复制 php.ini-dist 或 php.ini-recommended 到/usr/local/lib 目录，并命名为 php.ini：

```
cp php.ini-dist /usr/local/lib/php.ini
```





更改 httpd.conf 文件相关设置，该文件位于/usr/local/apache2/conf 中。找到该文件中的如下指令行：

```
AddType application/x-gzip .gz .tgz
```

在该指令后添加如下指令：

```
AddType application/x-httpd-php .php .phtml
```

重新启动 Apache，并在 Apache 主目录下建立文件 test.php：

```
<?php
	phpinfo();
?>
```

在浏览器中输入 http://127.0.0.1/test.php，按 Enter 键，如果出现如图 2.14 所示的页面，则 PHP 安装成功。



说明　Apache2 默认主目录为/usr/local/apache2/htdocs。

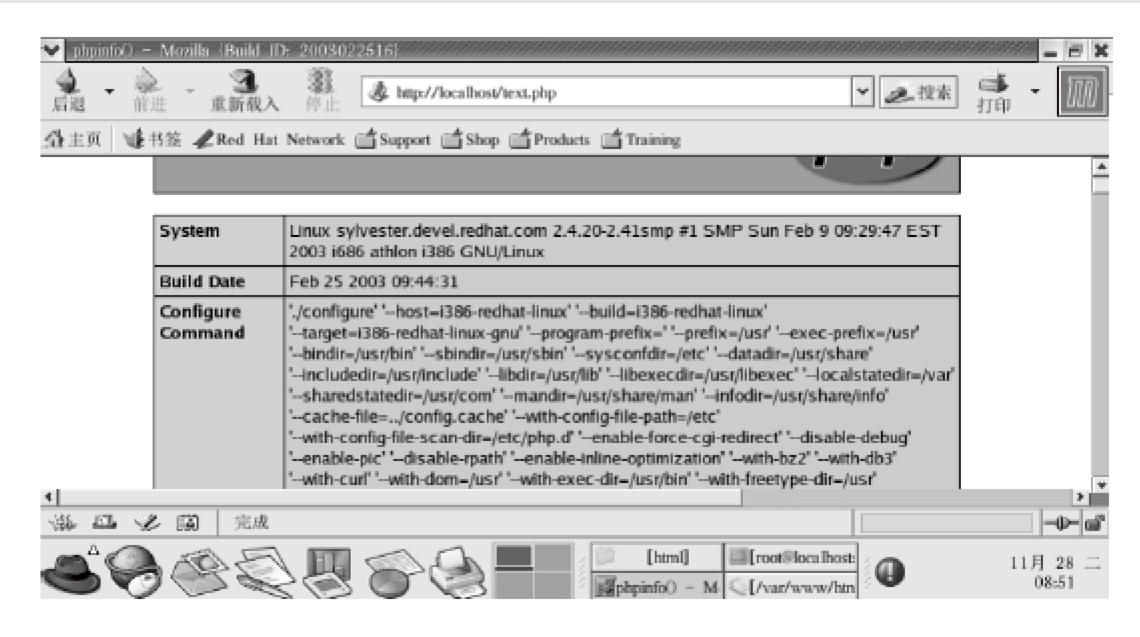


图 2.14　PHP 测试

技巧　安装文件的路径要遵循一定的客观原则，为了避免在 Windows 和 Linux 间移植程序时带来的不便，选择 D:\usr\local\php 的目录时要和在 Linux 下的安装目录相匹配。建议最好不要选择中间有空格的目录，如 E:\Program Files\PHP，这样做会导致一些未知错误或崩溃发生。

2.5　数据库设计

开发一个功能完善的网上社区离不开数据库的支持，只有拥有了强大的数据库，网上社区才能够存储大量的数据信息，实现更多、更好的功能来吸引更多的社区成员。本节将对 BCTY365 网上社区数据库的设计进行详细介绍。

2.5.1　数据库分析

BCTY365 网上社区是一个中型的面向软件开发者的程序，考虑到开发的成本、搭配的合理性以及

操作的灵活性等，使用了 MySQL 数据库。从成本考虑 MySQL 数据库是完全免费的，可以在网上免费下载；从匹配的角度讲，PHP 与 MySQL 数据库一直是公认的最佳搭档；MySQL 数据库不但可以在命令模式下进行操作，而且还配备了一些比较流行的图形化管理工具，如 phpMyAdmin 等，可以轻松地对 MySQL 数据库中的数据进行操作。

2.5.2　数据库概念设计

根据上述各节对 BCTY365 网上社区系统做的需求分析和系统设计，规划出 BCTY365 网上社区的实体关系 E-R 图。其中包括注册用户实体、发帖信息实体、回帖信息实体、订单信息实体和编程词典信息实体，其他还有一些辅助的实体，是对上述实体中内容的补充。由于涉及的实体较多，这里只对注册用户实体、发帖信息实体和订单信息实体的 E-R 图进行介绍。

1．注册用户实体

注册用户实体用于存储用户注册信息，包括编号、用户名、真实姓名、密码、邮箱、性别、电话、QQ 号码、家庭地址、访问次数、注册时间、最后一次登录时间、IP 地址、邮政编码、用户类型、密码提示问题、密码答案、真实密码、表情图、发帖次数属性。注册用户实体的 E-R 图如图 2.15 所示。

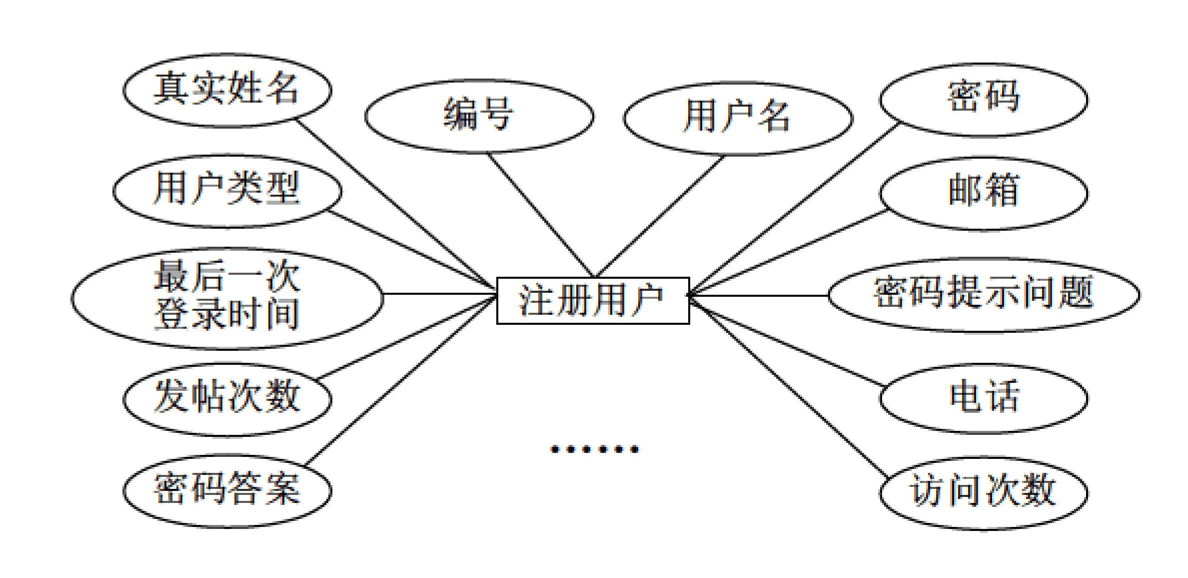


图 2.15　注册用户实体 E-R 图

2．发帖信息实体

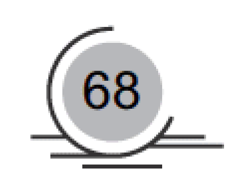
发帖信息实体用于存储登录本社区的会员在论坛中发布帖子的相关信息，包括编号、用户名 ID、帖子类别、帖子标题、帖子内容、发帖时间、最后回复时间、表情图、访问次数、是否顶帖、上传图片属性。发帖信息实体的 E-R 图如图 2.16 所示。

3．订单信息实体

订单信息实体存储用户在线购买时填写的订单信息，包括编号、用户名、性别、家庭地址、邮政编码、QQ 号码、邮箱、手机号码、电话号码、收货方式、邮资、产品金额、订单时间、订单号、选择城市等属性。订单信息实体的 E-R 图如图 2.17 所示。



图 2.16　发帖信息实体 E-R 图　　　　图 2.17　订单信息实体 E-R 图





2.5.3　创建数据库及数据表

在 BCTY365 网上社区系统中应用的是 db_bcty365 数据库，其中涉及 18 个数据表，数据表的名称和功能如图 2.18 所示。

本案例中创建数据库和数据表使用的是 phpMyAdmin 图形化管理工具，下面将介绍数据库和数据表的创建方法，以及在创建过程中需要注意的一些问题。

1．数据库的创建

打开 phpMyAdmin 图形化管理工具的主页，首先在文本框中输入要创建的数据库的名称（如 db_bcty365），然后在下拉列表框中选择要使用的字符编码格式，这里使用的是 gb2312_chinese_ci，如图 2.19 所示。最后单击“创建”按钮，数据库创建成功。

服务器: localhost ▸ 数据库: db_bcty365

表	类型	整理	说明
tb_bb	MyISAM	gb2312_chinese_ci	编程词典版本信息表
tb_bbqb	MyISAM	gb2312_chinese_ci	版本之间区别信息表
tb_bbs	MyISAM	gb2312_chinese_ci	论坛发帖信息表
tb_bccd	MyISAM	gb2312_chinese_ci	编程词典信息表
tb_city	MyISAM	gb2312_chinese_ci	城市信息表
tb_cjwt	MyISAM	gb2312_chinese_ci	常见问题信息表
tb_dd	MyISAM	gb2312_chinese_ci	订单信息表
tb_reply	MyISAM	gb2312_chinese_ci	论坛回帖信息表
tb_sjxz	MyISAM	gb2312_chinese_ci	软件升级下载信息表
tb_soft	MyISAM	gb2312_chinese_ci	软件下载信息表
tb_tell	MyISAM	gb2312_chinese_ci	社区公告信息表
tb_type	MyISAM	gb2312_chinese_ci	社区模块类型信息表
tb_type_big	MyISAM	gb2312_chinese_ci	论坛大类信息表
tb_type_small	MyISAM	gb2312_chinese_ci	论坛小类信息表
tb_user	MyISAM	gb2312_chinese_ci	注册用户信息表
tb_xlh	MyISAM	gb2312_chinese_ci	升级下载序列号信息表
tb_bccdjj	MyISAM	gb2312_chinese_ci	编程词典简介信息表
tb_leaveword	MyISAM	gb2312_chinese_ci	存储客户反馈信息

图 2.18　db_bcty365 数据库中使用的数据表

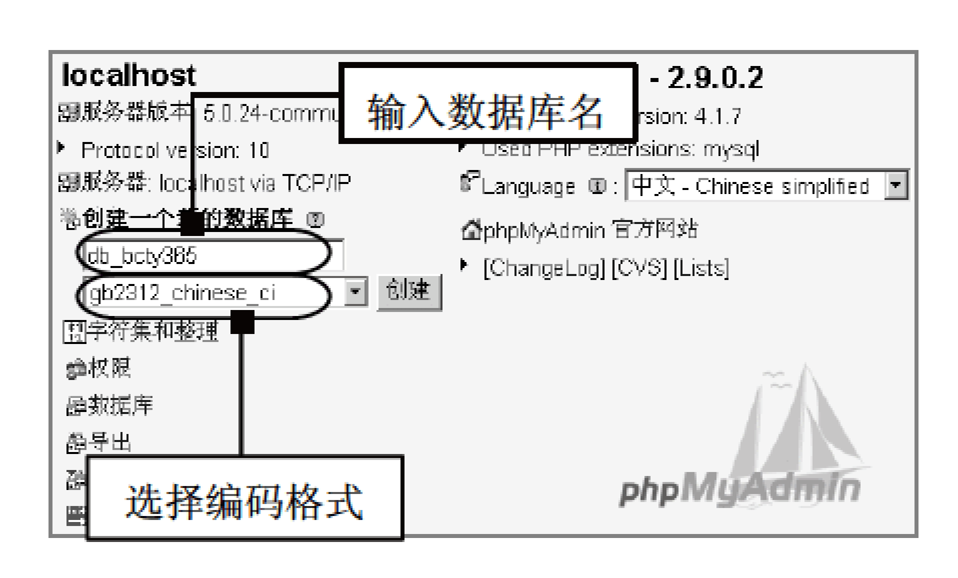


图 2.19　phpMyAdmin 管理界面

技巧　创建数据库的过程中，尽量使用与程序内容贴切的英文字符进行命名，有助于对数据库的理解。如果使用 AppServ 配置 PHP 开发环境，那么在创建数据库时不需要指定编码的格式，默认值为 gb2312_chinese_ci；如果自行配置开发环境，那么就要指定编码格式为 gb2312_chinese_ci，否则创建数据库的编码格式为 latin1_swedish_ci，将导致数据库中的数据出现乱码。

2．创建数据表

在成功创建数据库以后，接下来就是创建数据表，这里以 tb_bb 编程词典版本信息表为例，讲解如何创建数据表，以及在创建数据表的过程中都需要注意哪些问题。这里创建一个名字为 tb_bb 的数据表，包括 3 个字段，如图 2.20 所示。

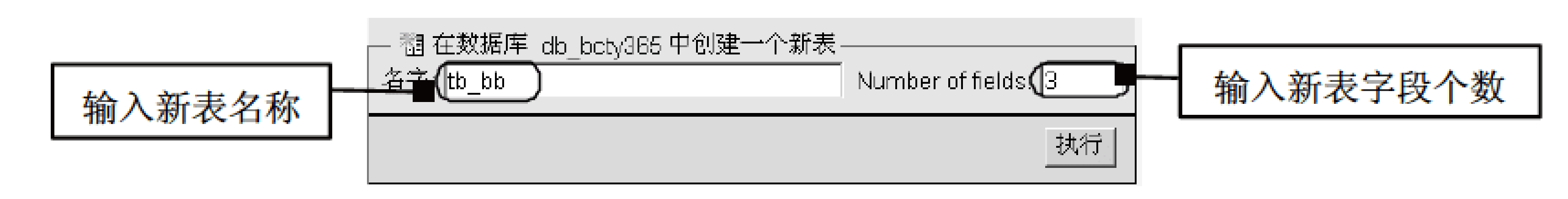


图 2.20　创建 tb_bb 数据表

单击“执行”按钮后，进入到如图 2.21 所示的添加字段信息的页面中，在此处对字段进行详细的



设置，包括字段名、数据类型、长度/值、属性、默认值、额外、主键和索引等。

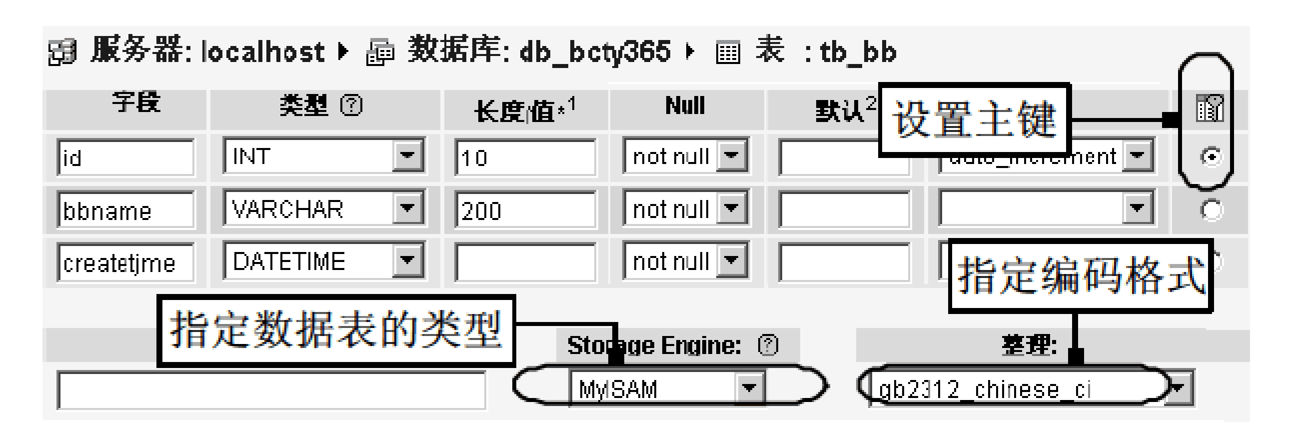


图 2.21　添加数据表中字段信息

技巧　创建数据库中的数据表时，字段名的设计尽量要与数据表的内容相符合，这样有助于程序后期维护和修改工作的进行，能够直观地看出数据表的作用。

如果使用 AppServ 配置 PHP 开发环境，那么在创建数据表时不需要指定数据表类型和编码格式；如果使用自行配置的 PHP 开发环境，那么就要指定数据表的类型为 MyISAM 和字符的编码格式为 gb2312_chinese_ci，否则创建的数据表类型为 InnoDB，而编码格式为 latin1_swedish_ci，将导致该数据表中的数据复制到其他机器上后不可用，并且数据表中的数据出现乱码。

在创建数据表的过程中，一定要为数据表指定一个主键，它是数据表的唯一标识。

掌握数据表的创建方法后，就可以自行创建数据表。由于本案例中涉及的数据表多达 18 个，这里不能对每个数据表的功能设计进行一一介绍，所以只给出几个重要的数据表的设计效果图供广大读者参考，其他数据表请参见本书附带的光盘。数据表的设计结构如图 2.22～图 2.24 所示。

（1）tb_user（注册用户信息表）

注册用户信息表主要用于存储本社区中会员的个人信息。该数据表的结构如图 2.22 所示。

（2）tb_reply（论坛回帖信息表）

论坛回帖信息表主要用于存储登录会员在本社区中回复帖子的信息。该数据表的结构如图 2.23 所示。

（3）tb_bccd（编程词典信息表）

编程词典信息表主要用于存储本社区的在线订购模块中出售的编程词典的基本信息。该数据表的结构如图 2.24 所示。

服务器: localhost ▸ 数据库: db_bcty365 ▸ 表 : tb_user

字段	类型	整理	属性	Null	默认	额外	说明
id	int(8)			否		auto_increment	自动编号ID
userne	varchar(50)	gb2312_chinese_ci		是	NULL		注册用户名
truename	varchar(50)	gb2312_chinese_ci		是	NULL		真实姓名
pwd	varchar(50)	gb2312_chinese_ci		是	NULL		注册密码
email	varchar(50)	gb2312_chinese_ci		是	NULL		有效邮箱地址
sex	varchar(2)	gb2312_chinese_ci		是	NULL		性别
tel	varchar(20)	gb2312_chinese_ci		是	NULL		联系电话
qq	varchar(20)	gb2312_chinese_ci		是	NULL		QQ号码
address	varchar(100)	gb2312_chinese_ci		是	NULL		联系地址
logintimes	int(8)			否			访问次数
regtime	datetime			否			注册时间
lastlogintime	datetime			否			最后一次登录时间
ip	varchar(20)	gb2312_chinese_ci		否			IP地址
yb	varchar(20)	gb2312_chinese_ci		是	NULL		邮政编码
usertype	int(2)			否			用户类型
question	varchar(200)	gb2312_chinese_ci		否			密码提示问题
answer	varchar(200)	gb2312_chinese_ci		否			密码提示答案
truepwd	varchar(200)	gb2312_chinese_ci		否			真实密码
photo	varchar(50)	gb2312_chinese_ci		否			表情图
pubtimes	int(4)			是	0		发帖次数

图 2.22　注册用户信息表结构

服务器: localhost ▸ 数据库: db_bcty365 ▸ 表 : tb_reply

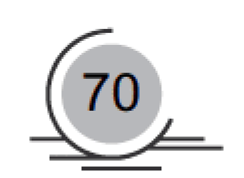
字段	类型	整理	属性	Null	默认	额外	说明
id	int(8)			否		auto_increment	自动编号ID
userid	int(8)			否	0		注册用户ID
bbsid	int(8)			否	0		发布帖子表ID
title	varchar(200)	gb2312_chinese_ci		是	NULL		回复主题
content	mediumtext	gb2312_chinese_ci		是	NULL		回复内容
createtime	datetime			是	NULL		回复时间
mark	int(2)			是	NULL		回复记录
photo	varchar(80)	gb2312_chinese_ci		是	NULL		回复图片

图 2.23　论坛回帖信息表结构

服务器: localhost ▸ 数据库: db_bcty365 ▸ 表 : tb_bccd

字段	类型	整理	属性	Null	默认	额外	说明
id	int(8)			否		auto_increment	自动编号ID
bccdname	varchar(200)	gb2312_chinese_ci		否			编程词典名称
owner	varchar(100)	gb2312_chinese_ci		否			开发者
typeid	varchar(50)	gb2312_chinese_ci		否			版本类型
content	mediumtext	gb2312_chinese_ci		否			编程词典简介
samepart	mediumtext	gb2312_chinese_ci		否			软件共同点
addtime	datetime			否			开发时间
imageaddress	varchar(100)	gb2312_chinese_ci		否			界面存储地址
bbid	int(8)			是	NULL		所属版本的ID
price	float			是	NULL		价格

图 2.24　编程词典信息表结构





2.6　公共模块设计

2.6.1　数据库连接文件

在进行程序开发的过程中，有很多地方都涉及数据库的应用，在应用数据库之前首先要与数据库建立连接，因此可以将数据库的连接代码作为一个公共文件进行存储，在需要使用数据库连接文件的地方直接调用该文件即可，无须重复编写相同的代码，既减少了代码的冗余，也便于对数据库连接文件进行修改。在本项目中笔者将数据库的连接代码存储于 conn.php 中。conn.php 文件的代码如下：

例程01　代码位置：光盘\TM\02\bcty365\conn\conn.php

```
<?php
❶    $conn=mysql_connect("localhost","root","root");  //连接数据库服务器
❷    mysql_select_db("db_bcty365",$conn);            //连接指定的数据库
❸    mysql_query("set names gb2312");                //对数据库中的编码格式进行转换，避免出现中文乱码的问题
?>
```

代码贴士

❶ mysql_conect()：连接 MySQL 服务器，服务器的用户名为 root，密码为 root。
❷ mysql_select_db()：用于连接指定的 MySQL 数据库，数据库名为 db_bcty365。
❸ set names gb2312：指定数据库中字符的编码格式为 gb2312。

成功创建 conn.php 文件后，在需要进行数据库操作的程序中，就可以通过 include 或者其他包含语句调用 conn.php 文件即可，无须再编写连接数据库的程序代码。应用 include 语句包含 conn.php 文件的代码如下：

```
<?php
    include ("conn/conn.php");                  //包含数据库文件
?>
```

2.6.2　将文本中的字符转换为 HTML 标识符

在输出数据库中数据的过程中，有必要将数据中的一些特殊字符转换为 HTML 标识符，这样可以避免一些不必要的麻烦。例如，在输出一个程序的执行代码的过程中，如果不对其进行转换，那么输出的将不是程序的代码，而是程序的执行结果。这里将文本中字符的转换编写到一个自定义函数 unhtml()中，并保存到 function.php 文件中，将其作为一个公共模块来使用，当需要使用时直接调用 function.php 文件即可。function.php 文件包含两个自定义函数，即 unhtml()函数和 msubstr()函数，unhtml()函数用于将数据中的特殊字符转换为 HTML 标识符；msubstr()函数用于对字符串进行指定长度的截取。代码如下：

例程02　代码位置：光盘\TM\02\bcty365\function.php

```
<?php
function unhtml($content){                      //定义自定义函数的名称
```



```
❶ $content=htmlspecialchars($content);                 //转换文本中的特殊字符
❷ $content=str_replace(chr(13),"<br>",$content);      //替换文本中的换行符
   $content=str_replace(chr(32)," ",$content);    //替换文本中的“ ”
   $content=str_replace("[_[","<",$content];           //替换文本中的大于号
   $content=str_replace(")_)",">",$content);           //替换文本中的小于号
   $content=str_replace("|_|"," ",$content);           //替换文本中的空格
❸ return trim($content);                               //删除文本中首尾的空格
}
//定义一个用于截取一段字符串的函数 msubstr()
function msubstr($str,$start,$len){     //$str 指的是字符串，$start 指的是字符串的起始位置，$len 指的是长度
$strlen=$start+$len;                    //用$strlen 存储字符串的总长度（从字符串的起始位置到字符串的总长度）
for($i=0;$i<$strlen,$i++){              //通过 for 循环语句，循环读取字符串
❹  if(ord(substr($str,$i,1))>0xa0){    //如果字符串中首个字节的 ASCII 序数值大于 0xa0，则表示为汉字
    $tmpstr.=substr($srt,$i,2);         //每次取出两位字符赋给变量$tmpstr，即等于一个汉字
    $i++;                               //变量自加 1
    }else{                              //如果不是汉字，则每次取出一位字符赋给变量$tmpstr
         $tmpstr.=substr($str,$i,1);}
    }
return $tmpstr;                         //输出字符串
}?>
```

代码贴士

❶ htmlspecialchars()：将特殊字符转换成 HTML 格式，而不会将所有字符都转换成 HTML 格式。

❷ str_replace(mixed search,mixed replace,mixed subject,int cont)：将所有在参数 subject 中出现的 search 以参数 replace 替换，参数&count 表示替换字符串执行的次数。

❸ trim()：删除字符串中首尾的空白或者其他字符。

❹ substr()：从指定的字符串中按照指定的位置截取一定长度的字符。

2.7　前台首页设计

当今时代，很多人都十分重视事物的第一印象，第一印象基本上就决定对某个事物的看法和态度，在网络中更是如此，网站给人的第一印象如果不好，那么就会有很多人因此而不去浏览该网站，无论网站的内容是否丰富。可以说网站首页设计的成功与否直接影响着整个网站的发展。

2.7.1　前台首页概述

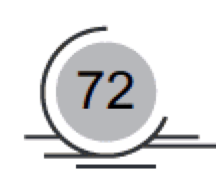
网站首页是整个网站的脸面，既要突出企业的形象，又要展示出网站强大的功能。如果网站首页设计得非常成功，那无疑是为整个网站的成功增添了一个砝码。BCTY365 网上社区首页的设计以企业的品牌形象为基础，全力打造网站的整体功能，重点推出企业的软件产品，具体内容如下。

☑ 网站菜单导航：包括“首页”、“技术支持”、“在线订购”、“社区论坛”、“软件下载”、“升级下载”、“购买须知”、“联系我们”等导航菜单。

☑ 用户注册和登录模块：实现用户注册、会员登录、找回密码和修改密码的功能。

☑ 网站公告：主要用于发布社区中的一些新消息和重大事件。

☑ 编程词典模块：推广企业的软件产品。





☑ 软件下载模块：展示企业提供的适用版和免费的软件产品。

☑ 常见问题模块：列举出编程中常见问题的解决方案。

☑ 社区论坛模块：浏览社区论坛中的部分帖子。

☑ 升级下载模块：提供一些软件的升级版本下载。

上述内容就是 BCTY365 网上社区首页中体现出的内容，为了更加直观地了解网上社区首页的设计，这里先预览一下社区首页，该首页在本书光盘中的路径为\TM\02\bcty365\index.php，如图 2.25 所示。

BCTY365 网上社区首页的设计看上去很复杂，由多个版块组成，但实现的过程非常简单。总体架构使用一个 2 行 3 列的表格和一个 3 行 3 列的表格，将其分割成不同的版块，然后使用脚本语句从数据库中读取数据，最后将数据循环输出到页面中，其中网站的头尾文件使用 include 包含语句调用。首页的框架结构如图 2.26 所示。

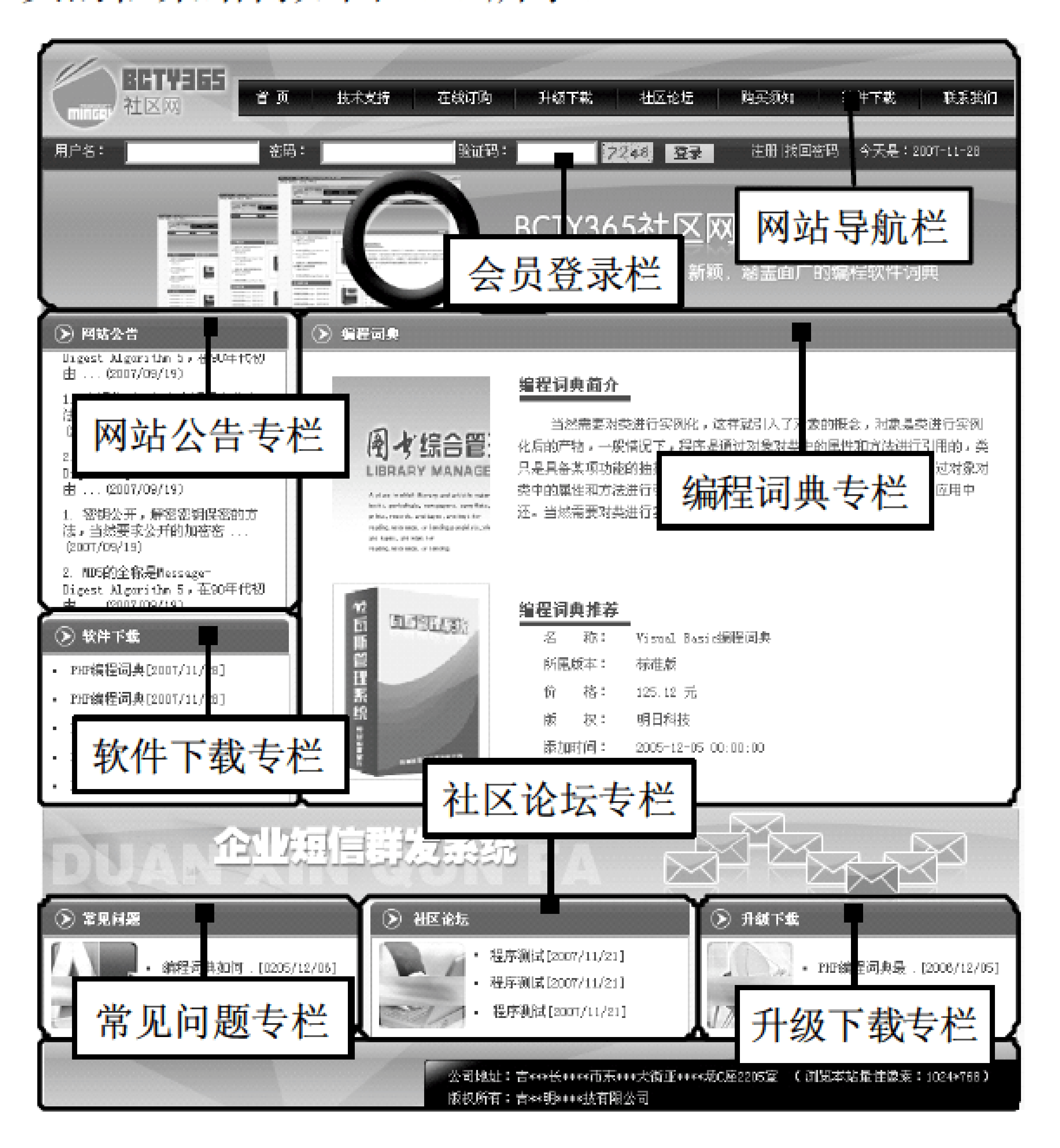


图 2.25　BCTY365 网上社区首页



图 2.26　网站首页的框架结构

2.7.2　前台首页技术分析

作为网站首页，不一定要具有什么特殊的技术或者功能，应该是以简洁、鲜明、突出企业形象、展示网站的功能为主。即使使用的是一个静态页面，只要能够将内容表达全面、完整，那么这个首页设计也是非常成功的。

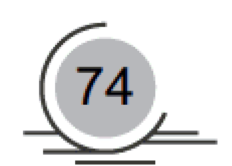
在本案例首页的设计中，应用到一个文字循环滚动的技术，通过该技术来输出社区中发布的公告信息。该技术是通过 JavaScript 脚本和 div 标签来共同实现的，其实现的原理是：首先创建一个 div 标签，然后在 div 标签中输出公告信息，最后通过 JavaScript 来对 div 标签进行操作，实现 div 标签中内容的滚动输出。该技术的实现在 index.php 页中完成，其中使用的 JavaScript 脚本的代码如下：

例程03　代码位置：光盘\TM\02\bcty365\index.php

```
<script language="JavaScript">
    marqueesHeight=222;                                    //定义输出标签的高度
    stopscroll=false;                                      //定义 stopscroll 的默认值为 false
    with(marquees){                                        //编辑 marquees 标签的属性
        style.width=0;                                     //定义初始宽为 0
        style.height=marqueesHeight;                       //定义 marquees 标签的高为 222
        style.overflowX="visible";                         //定义值为显示
        style.overflowY="hidden";                          //定义值为隐藏
        noWrap=true;
        onmouseover=new Function("stopscroll=true");       //当鼠标经过时执行 stopscroll=true
        onmouseout=new Function("stopscroll=false");       //当鼠标离开时执行 stopscroll=false
    }
    //创建一个新的 div"templayer"与 div"marquees"进行连接，实现不间断的循环输出内容
    document.write('<div id="templayer" style="position:absolute;z-index:1;visibility:hidden"></div>');
    preTop=0; currentTop=0;
    function init(){
        templayer.innerHTML="";                            //设置 templayer 的初始值为空
        while(templayer.offsetHeight<marqueesHeight){      //判断当 templayer 的高度小于 marquees 的高度时
            templayer.innerHTML+=marquees.innerHTML; //将 templayer 的值赋给 marquees
        }
        marquees.innerHTML=templayer.innerHTML+templayer.innerHTML;//将 templayer 的值累加
        setInterval("scrollup()",50);                      //间隔 50 毫秒执行一次 scrollup()函数
    }
    function scrollup(){                                   //实现滚动输出
        if(stopscroll==true) return;                       //判断如果 stopscroll==true，不执行循环
        preTop=marquees.scrollTop;
        marquees.scrollTop+=1;
        if(preTop==marquees.scrollTop){
            marquees.scrollTop=templayer.offsetHeight-marqueesHeight;
            marquees.scrollTop+=1;
        }
    }
</script>
```

在 div 标签中，主要是输出数据库中存储的公告信息，并且对输出的信息进行截取和替换，规范输出的内容。div 标签中的程序代码如下：

```
<div id=marquees > <!--创建一个 div 标签-->
<table width="200" height="25" border="0" align="center"  cellpadding="0" cellspacing="0">
    <?php                                                  //从数据库中读取公告数据
        $sql=mysql_query("select id,title,createtime from tb_tell order by createtime desc limit 0,10",$conn);
        $info=mysql_fetch_array($sql);
        if($info==false){                                  //判断当$info==false 时执行下面的内容
    ?>
    <tr>
        <td height="25"><div align="center"><a href="#" class="a4">本站暂无公告发布! </a></div></td>
   </tr>
    <?php    }else{
        $i=1;                                              //定义变量$i=1
```





```
            do{                                                          //执行 do…while 循环语句
        ?>
        <tr>
            <td height="25" style="padding:4">
            <a href="tellinfo.php?id=<?php echo $info[id];?>" class="a1">
        <?php
            if($i==1){                                                   //判断当$i==1 时，将输出的内容设置为红色
                echo "<font color=red>";
            }
            echo $i.". ";
            echo unhtml(msubstr($info[title],0,50));                     //应用自定义函数对输出的内容进行控制
            if(strlen($info[title])>50){                                 //当输出内容的长度超过 50 个字符时用“...”代替
                echo " ...";
            }
            echo "(".str_replace("-","/",$info[createtime]).")";//将输出的公告时间中的“-”用“/”替代
            if($i==1){  echo "</font>"; }
        ?>        </a>
            </td>
        </tr>
        <?php
            $i++;
            }while($info=mysql_fetch_array($sql));      //do…while 循环语句结束
            }
        ?>
</table>
    </div>
```

在首页中使用滚动条是一个比较不错的方法，可以增加网页的动态效果，增加网页的观赏性，而且不会影响到网页的浏览速度。

2.7.3　前台首页的实现过程

前台首页使用的数据表：tb_tell、tb_soft、tb_bccd、tb_bb、tb_cjwt、tb_bbs、tb_sjxz

开发网站首页主要就是连接数据库，从数据库中读取数据，最后应用循环语句将数据库中的数据输出到前台页面。由于使用的代码较多，而且多数都是重复使用，所以这里只给出首页中公告发布模块的代码。

公告发布模块主要实现从数据库中读取公告数据，将数据在首页中滚动输出，并且对公告信息的长度进行控制，保证内容的整齐、规范。详细代码可以参考本书光盘中的 TM\02\bcty365\index.php 文件。index.php 文件的部分代码如下：

例程04　代码位置：光盘\TM\02\bcty365\index.php

```
<?php include_once("top.php");                              //获取头部文件>
…//省略部分代码
<?php
    //读取数据库中公告表中的数据
    $sql=mysql_query("select id,title,createtime from tb_tell order by createtime desc limit 0,10",$conn);
    $info=mysql_fetch_array($sql);                          //执行读取数据表中数据的语句
    if($info==false){                                       //如果返回值为空则执行下面的语句
```

```
?>
<tr>
    <td height="25"><div align="center"><a href="#" class="a4">本站暂无公告发布! </a></div></td>
</tr>
<?php
    }else{                                   //如果返回值不为空，则执行下面的 do…while 循环语句
        $i=1;
        do{
?>
<tr>
    <td height="25" style="padding:4"><a href="tellinfo.php?id=<?php echo $info[id];?>" class="a1">
<?php
    if($i==1){                               //判断当变量$i==1 时，输出的内容以红色字体显示
        echo "<font color=red>";
    }
    echo $i.". ";
❶  echo unhtml(msubstr($info[title],0,50));
❷  if(strlen($info[title])>50){              //如果标题长度大于 50 个字符，则以省略号代替
        echo " ...";
    }
❸  echo "(".str_replace("-","/",$info[createtime]).")"; //输出公告发布的时间，并且将其中的“/”使用“-”替换
    if($i==1){
        echo "</font>";
    }
?>
        </a>
    </td>
</tr>
<?php
$i++;                                        //变量自加 1
}while($info=mysql_fetch_array($sql));       //do…while 循环语句结束
}
?>
…                                            //省略了部分代码
<?php
include_once("bottom.php");                  //调用网站的尾文件
?>
```

代码贴士

❶ 应用 unhtml()和 msubstr()自定义函数去除输出字符串中的空格和控制输出字符串的长度。

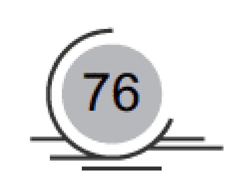
❷ strlen()：获取指定字符串的长度。

❸ str_replace()：实现字符串的替换。

2.8 注册模块设计

2.8.1 注册模块概述

BCTY365 网上社区系统为了更好地与广大网民朋友进行交流和沟通，创建了一个会员注册模块。



通过会员注册模块，可以有效地对用户信息进行采集，并将合法的用户信息保存到指定的数据表中，实现与用户的长期沟通和交流。既然设置了会员注册模块，那么在系统中就要为会员提供一些特殊的权限。在本系统中注册会员可以拥有如下权限：在本社区的论坛中发布和回复帖子、在技术支持模块中发表留言、在升级下载模块中下载软件升级包等，而且可以进行修改密码和找回密码。会员注册模块的运行结果如图 2.27 所示。



图 2.27　用户注册模块的运行结果

2.8.2　注册模块技术分析

在会员注册模块中，必不可少的就是要对用户输入的信息进行判断，首先判断用户填写的注册信息中哪些是必须填写的，哪些可以不填写，然后进一步判断输入的信息是否合理合法，例如，判断输入邮编的格式是否正确，判断输入邮箱的格式是否正确等。对表单中提交的数据进行判断最常用的办法就是使用 JavaScript 脚本，也可以使用正则表达式。下面讲解在本模块中如何通过 JavaScript 实现表单提交数据验证。

操作原理是：在 form 表单中调用 onsubmit 事件，通过该事件调用指定的 JavaScript 脚本，执行 chkinput()自定义函数，实现对表单中提交数据的验证。在 JavaScript 脚本中，实现对表单中提交的数据进行判断，判断输入的内容是否为空以及格式是否正确，如果正确则继续执行，否则将弹出提示对话框，并将鼠标的焦点指定到出错的位置。具体的 JavaScript 脚本代码如下：

```
<script language="JavaScript" type="text/javascript">
    function chkinput(form){                                //定义一个函数
        if(form.tel.value==""){                             //判断 tel 文本框中的值是否为空
            alert("请填写联系电话!");                        //如果为空，则输出“请填写联系电话!”
            form.tel.select();                              //返回到 tel 文本框
            return(false);
        }
        if(form.email.value==""){                           //判断 email 文本框的值是否为空
            alert("请输入 E-mail 地址!");                    //如果为空，则输出“请输入 E-mail 地址!”
          form.email.select();                              //返回到 email 文本框
          return(false);
        }
        var i=form.email.value.indexOf("@");
        var j=form.email.value.indexOf(".");
        //进一步判断邮箱的格式是否正确，是否包含“@”和“.”
        if((i<0)||(i-j>0)||(j<0)){
            alert("请输入正确的 E-mail 地址!");
            form.email.select();                            //返回到 email 文本框
            return(false);
        }
      return(true);                                         //提交表单
    }
</script>
```

上述代码中，只是列举了 JavaScript 中的部分内容，并且在对电话号码进行判断时，只是判断其是否为空，没有进一步判断电话号码的格式是否正确。如果想要更加准确地判断电话号码的格式是否正确，可以采用下面的方法：通过正则表达式的 preg_match()函数，在表单提交处理页中对电话号码进行判断。

preg_match()函数的语法格式如下：

```
int preg_match ( string pattern, string subject [, array matches [, int flags]] )
```

preg_match()函数的参数说明如表 2.3 所示。

表 2.3　preg_match()函数的参数说明

参　　数	说　　明
pattern	必要参数。需要匹配的正则表达式
subject	必要参数。输入的字符串
matches	可选参数。输出的搜索结果的数组，如$out[0]将包含与整个模式匹配的结果，$out[1]将包含与第一个捕获的括号中的子模式所匹配的结果，依次类推
flags	可选参数。标记 PREG_OFFSET_CAPTURE 可对每个出现的匹配结果也同时返回其附属的字符串偏移量，本标记自 PHP 4.3.0 起可用

通过 preg_match()函数判断电话号码的格式是否正确的方法如下：首先定义一个用于判断电话号码格式的正则表达式。代码如下：

```
/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$/
```

正则表达式的功能分析如下：使用“^”和“$”对字符串进行边界的限制，对区号从字符串的开始进行匹配，对其他号码从字符串的末尾开始进行匹配；将括号“()”中的内容作为一个原子使用；使用“\d”来匹配一个数字，区号为 3 或 4 个数字，其他数字为 7 或 8 个；使用“{}”来对前字符进行重复匹配；使用“|”对匹配的模式进行选择，分成 3 个模式。

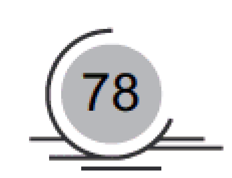
然后将该正则表达式应用到 preg_match()函数中，对表单提交的电话号码进行判断，如果正确则继续执行，否则弹出提示信息，并返回到表单提交页。代码如下：

```
<?php
   $tel="0431-84978981";                                    //定义一个电话号码的变量
   if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$tel,$counts)){
        echo "您输入的电话号码格式正确!";                      //输出字符串
   }else{
        echo "<script>alert('您输入的电话号码的格式不正确!!');history.back()</script>";
   }
?>
```

2.8.3　注册模块的实现过程

注册模块使用的数据表：tb_user

注册模块的实现过程非常简单，首先阅读注册服务条款，然后填写用户注册的用户名和密码，提



交后由系统判断输入的用户名是否被占用，如果未被占用则可以继续注册，填写详细的注册信息，将数据提交到表单处理页进行处理，最后将用户注册的信息保存到指定的数据表中。用户注册模块的实现过程主要由 3 个文件完成：register.php 用于输出注册服务条款，以及填写注册的用户名和密码，并且判断注册的用户名和密码是否被占用；getuserinfo.php 文件用于填写详细的注册信息，并且在表单中应用数字验证码技术；savereginfo.php 文件用于对表单中提交的数据进行处理，将数据保存到指定的数据表中。

在 savereginfo.php 文件中，首先连接数据库，然后获取表单中提交的数据，并且判断提交的用户名是否被占用，最后将提交的数据进行处理，并将数据保存到指定的数据表中。程序代码如下：

例程05　代码位置：光盘\TM\02\bcty365\saverreginfo.php

```
<?php session_start();                                          //初始化 session 变量
include_once("conn/conn.php");                                  //连接数据库
$usernc=trim($_POST[usernc]);                                   //获取注册的用户名
//判断指定的用户名是否存在
$sql=mysql_query("select usernc from tb_user where usernc='".$usernc."'",$conn);
$info=mysql_fetch_array($sql);                                  //按指定条件检索数据信息
if($info!=false){                                               //如果查询结果不为空，则执行以下操作
      echo "<script language='javascript'>alert('对不起，该昵称已被其他用户使用!');history.back();</script>";
      exit;
}
$xym=trim($_POST[xym]);                                         //去除变量两边的空格
$num=$_POST[num];                                               //接收变量值
if(strval($xym)!=strval($num)){
      echo "<script>alert('验证码输入错误!');window.location.href='register.php';</script>";
      exit;}
//对表单提交的数据进行处理
$truepwd=trim($_POST[pwd1]);                                    //获取真实密码
$pwd=md5($truepwd);                                             //获取加密密码
$truename=trim($_POST[truename]);                               //获取真实姓名
$email=trim($_POST[email]);                                     //获取邮箱地址
$sex=$_POST[sex];                                               //获取性别
$tel=trim($_POST[tel]);                                         //获取电话
$yb=trim($_POST[yb]);                                           //获取邮政编码
$qq=trim($_POST[qq]);                                           //获取 QQ
$address=trim($_POST[address]);                                 //获取地址
$question=trim($_POST[question]);                               //获取提示问题
$answer=trim($_POST[answer]);                                   //获取问题答案
$ip=getenv("REMOTE_ADDR");                                      //获取客户端的 IP
$logintimes=1;                                                  //指定访问次数
$regtime=date("Y-m-j H:i:s");                                   //获取时间
$lastlogintime=$regtime;
$usertype=0;                                                    //指定用户类型，默认为 0
$photo=$_POST["photo"];                                         //获取头像
//将表单中提交的数据存储到数据表中
if(mysql_query("insert into
tb_user(usernc,truename,pwd,email,sex,tel,qq,address,logintimes,regtime,lastlogintime,ip,usertype,yb,question
,answer,truepwd,photo)
values('$usernc','$truename','$pwd','$email','$sex','$tel','$qq','$address','$logintimes','$regtime','$lastlogintime','
$ip','$usertype','$yb','$question','$answer','$truepwd','$photo')",$conn)){
```

```
        session_register("unc");
        $_SESSION["unc"]=$usernc;
        echo "<script>alert('注册成功!');window.location.href='index.php';</script>";
}else{                                                          //如果添加操作失败，则给出提示
        echo "<script language='javascript'>alert('对不起,注册失败!');history.back();</script>";
        exit;                                                   //退出
}
?>
```

2.9　技术支持模块设计

技术支持模块主要是从浏览者的角度进行设计，存储大量技术问题的解决方案数据，为浏览者查阅提供方便，而且设计一个企业与客户沟通的平台，能够随时了解客户或者会员的意见和需求。

2.9.1　技术支持模块概述

技术支持模块主要由 3 个子模块组成，包括常见问题、客户反馈和联系方式。常见问题模块主要用于展示编程中一些常见问题的解决方案或者方法，为浏览者解决编程中的疑难问题提供方便；客户反馈模块主要用于收集和获取来自客户的需求和意见；联系方式模块主要用于展示企业的形象和具体的联系方式。

2.9.2　技术支持模块技术分析

技术支持模块中在对常见问题解决方案的数据进行输出时，使用了分页处理技术，该技术的设计思路是：从数据库中读取数据，获取数据总量，在每页中显示 20 条数据，根据数据总量和每页显示的条数对数据进行分页处理，计算出有多少页和当前显示的页码，实现首页、上一页、下一页和尾页之间的页面跳转。具体的设计思路可以参考 cjwt.php 文件中的代码注释和代码贴士。cjwt.php 文件的程序代码如下：

例程06　代码位置：光盘\TM\02\bcty365\cjwt.php

```
<?php
        $sql=mysql_query("select count(*) as total from tb_cjwt",$conn);   //读取数据库中的数据
        $info=mysql_fetch_array($sql);                                   //返回数据
❶      $total=$info[total];
//判断字段 total 是否为空，为空则执行下面的内容
        if($total==0){
?>
        <tr>
        <td height="22" colspan="2"><div align="center">对不起，暂无常见问题!</div></td>
       </tr>
<?php
        }else{                                                  //如果不为空，则执行下面的内容
❷              if(!isset($_GET["page"]) || !is_numeric($_GET["page"])){   //判断$_GET 获取的 page 的值是否存在
❸                      $page=1;                                 //如果不存在，则设置变量的值为 1
```

```
            }else{
❹                $page=intval($_GET["page"]);                  //如果存在，则获取变量$_GET 的值
            }
            //设置变量$pagesize,每页显示的数据数据量为 20
❺           $pagesize=20;
            if($total%$pagesize==0){                          //如果变量的值为 0
❻                $pagecount=intval($total/$pagesize);         //获取变量的整数值
            }else{
❼                $pagecount=ceil($total/$pagesize);           //如果不为 0，则获取实际的整数值
            }
            //读取数据库中的数据，按照时间进行降幂排列
            $sql=mysql_query("select * from tb_cjwt order by createtime desc limit ".($page-1)*$pagesize.",$pagesize
",$conn);
            while($info=mysql_fetch_array($sql)){
?>
…                                                             //省略部分代码
<table width="600" height="25" border="0" align="center" cellpadding="0" cellspacing="0">
    <tr>
            <td width="479"><div align="left">  共有常见问题 <?php echo $total;?> 条
 每页显示 <?php echo $pagesize;?> 条 第 <?php echo $page;?> 页/共
 <?php echo $pagecount;?> 页</div></td>
           <td width="269"><div align="right">
❽                <a href="<?php echo $_SERVER["PHP_SELF"]?>?page=1" class="a1">首页</a> 
                <a href="<?php echo $_SERVER["PHP_SELF"]?>?page=<?php
                    if($page>1)                                //判断如果页码大于 1
                        echo $page-1;                          //输出前一页
                    else
                        echo 1;
                        ?>" class="a1">上一页</a> 
                <a href="<?php echo $_SERVER["PHP_SELF"]?>?page=<?php
                if($page<$pagecount)                           //如果页码小于总页数
                    echo $page+1;
                else
                    echo $pagecount;                           //输出下一页
                        ?>" class="a1">下一页</a> 
                <a href="<?php echo $_SERVER["PHP_SELF"]?>?page=<?php echo $pagecount;?>"
class="a1">尾页</a>  </div>
            </td>
        </tr>
    </table>
```

代码贴士

❶ $total：为数据库中数据总的记录数。

❷ isset()用于检测变量是否已经设置；is_numeric()用于检测变量是否为数字或者数字字符串。

❸ $page：变量为页码中的第几页。

❹ intval()：获取变量的整数值。

❺ $pagesize：表示在每页中显示多少条数据。

❻ $pagecount：表示所有的数据可以分成多少页。

❼ ceil()：获取变量中的整数值，这里用于获取页码的整数值。

❽ $_SERVER["PHP_SELF"]：服务器变量，这里用于获取网页的链接地址。

2.9.3　常见问题的实现过程

常见问题使用的数据表：tb_cjwt

常见问题子模块实现的主要功能是展示出数据库中存储的有关编程中遇到的常见问题及解决方案。其运行结果如图 2.28 所示。



图 2.28　常见问题模块的运行结果

该模块由两个文件组成，一个是 cjwt.php 文件，用于存储创建问题数据，详细内容可以参考 2.9.2 节；另一个是 lookcjwt.php 文件，用于输出 cjwt.php 文件中对应问题的详细介绍和解决方案。代码如下：

例程07　代码位置：光盘\TM\02\bcty365\lookcjwt.php

```
<?php
include_once("conn/conn.php");                                    //与数据库建立连接
include_once("function.php");                                     //调用自定义函数
//读取 tb_cjwt 表中的数据，条件为 id="$_GET[id]"
$sql=mysql_query("select * from tb_cjwt where id='".$_GET["id"]."'",$conn);
$info=mysql_fetch_array($sql);
?>
<table width="635" height="100" border="0" align="center" cellpadding="0" cellspacing="0">
    <tr>
        <td width="94" height="30"><div align="center"><strong>问  题：</strong></div></td>
        <td width="541"><?php echo unhtml($info["question"]);         //输出问题的详细内容 ?></td>
    </tr>
    <tr>
        <td height="70"><div align="center"><strong>解  答：</strong></div></td>
        <td height="70"> <?php echo unhtml($info["answer"]);     //输出问题的解决方案 ?></td>
    </tr>
</table>
```

2.9.4　客户反馈的实现过程

客户反馈使用的数据表：tb_user、tb_leaveword

客户反馈子模块为客户提供了一个反馈意见和提出要求的平台，并且将提交的信息存储到数据库中。其运行结果如图 2.29 所示。

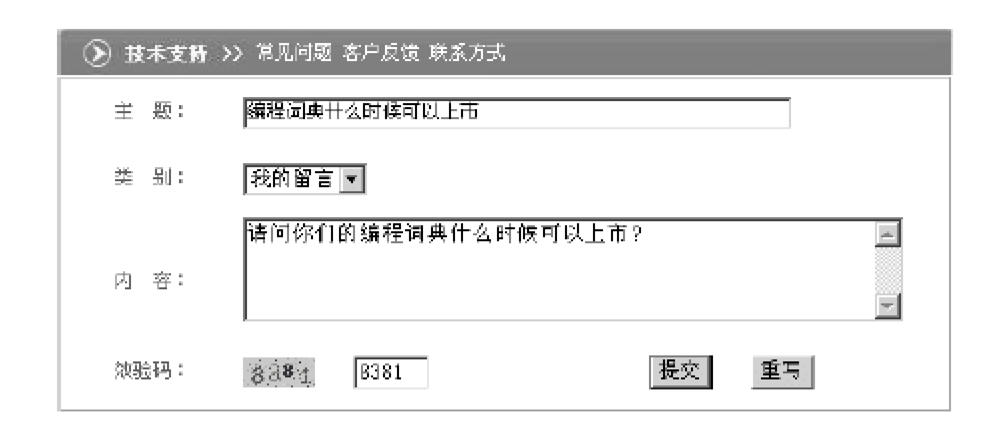
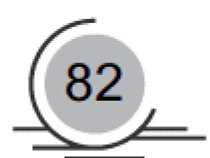


图 2.29　客户反馈模块的运行结果

该功能只对本网站中的会员开通，即只有以会员身份登录的用户才具有反馈信息的权限，其中在对提交表单的细节处理上使用 JavaScript 脚本来验证表单中的值是否为空，而且还使用了数字验证码技术。对表单中提交的数据进行处理是在 saveleaveword.php 文件中完成的，该文件主要用于对表单中提交的数据进行处理，将数据存储到数据库中。代码如下：



例程08　代码位置：光盘\TM\02\bcty365\saveleaveword.php

```
<?php session_start();                                                    //初始化一个 session 变量
$xym=$_POST[xym];                                                         //获取$_POST 提交的值
if($xym!=$_SESSION["autonum"]){                                           //判断验证码是否正确
     echo "<script>alert('效验码输入错误!');history.back();</script>";
     exit;
}
$title=$_POST["title"];                                                   //获取反馈信息的标题
$content=$_POST["content"];                                               //获取反馈信息的内容
$type=$_POST["type"];                                                     //获取反馈信息的类型
include_once("conn/conn.php");                                            //与数据库建立连接
$sql=mysql_query("select id from tb_user where usernc='".$_SESSION["unc"]."'",$conn); //读取数据库中的数据
$info=mysql_fetch_array($sql);                                            //获取结果集中的数组
$userid=$info["id"];
//向数据库中添加数据
if(mysql_query("insert into tb_leaveword(userid,type,title,content,createtime) values('$userid','$type','$title', '$content',
'".date("Y-m-j H:i:s")."')",$conn)){
     echo "<script>alert('留言发表成功!');history.back();</script>";        //添加操作成功，给出提示信息
}else{
     echo "<script>alert('留言发表失败!');history.back();</script>";        //添加操作失败，给出提示信息
}
?>
```

2.9.5　单元测试

在对客户反馈模块进行测试的过程中，当提交反馈信息时，提示“留言发表失败！”。经过初步的判断，问题可能出现在向数据库添加数据的过程中。为进一步查找出问题的根源，在 saveleaveword.php 页中应用了 mysql_error()函数，该函数可获取详细的错误信息，将它放置在 insert 添加语句之后，然后重新运行网页，运行结果如图 2.30 所示。

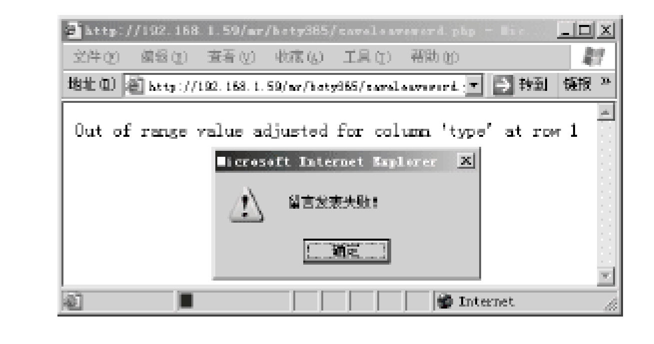


图 2.30　应用 mysql_error()函数获取的错误信息

在该结果图中可以看出是数据表中的 type 字段出了问题，经过仔细检查发现，原来是在编写 insert 添加语句的过程中，将语句中字段的排列顺序与对应的字段值的顺序弄错了，如图 2.31 所示。

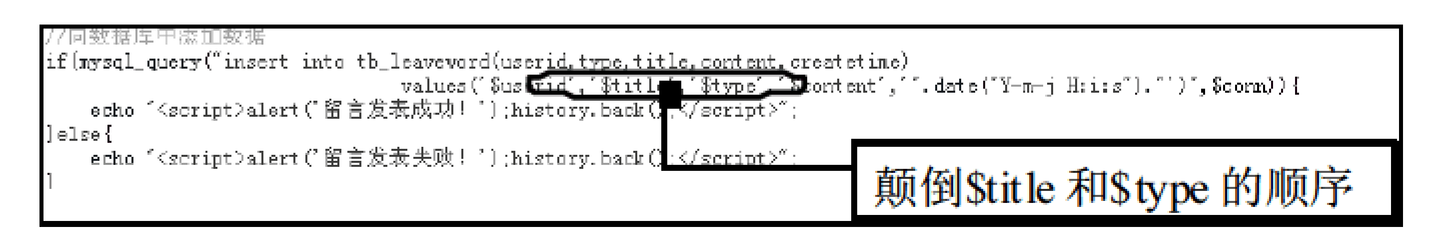


图 2.31　代码中的错误截图

调整变量$title 和$type 的顺序，重新运行程序，提示“留言发表成功！”。

2.10　在线订购模块设计

在线订购模块的功能是实现在线购买企业推出的软件产品，其操作的流程主要通过购物车和订单

管理来实现。

2.10.1　在线订购模块概述

在线订购的功能对所有访问网站的人开放，没有任何的权限限制。其操作流程如图 2.32 所示。

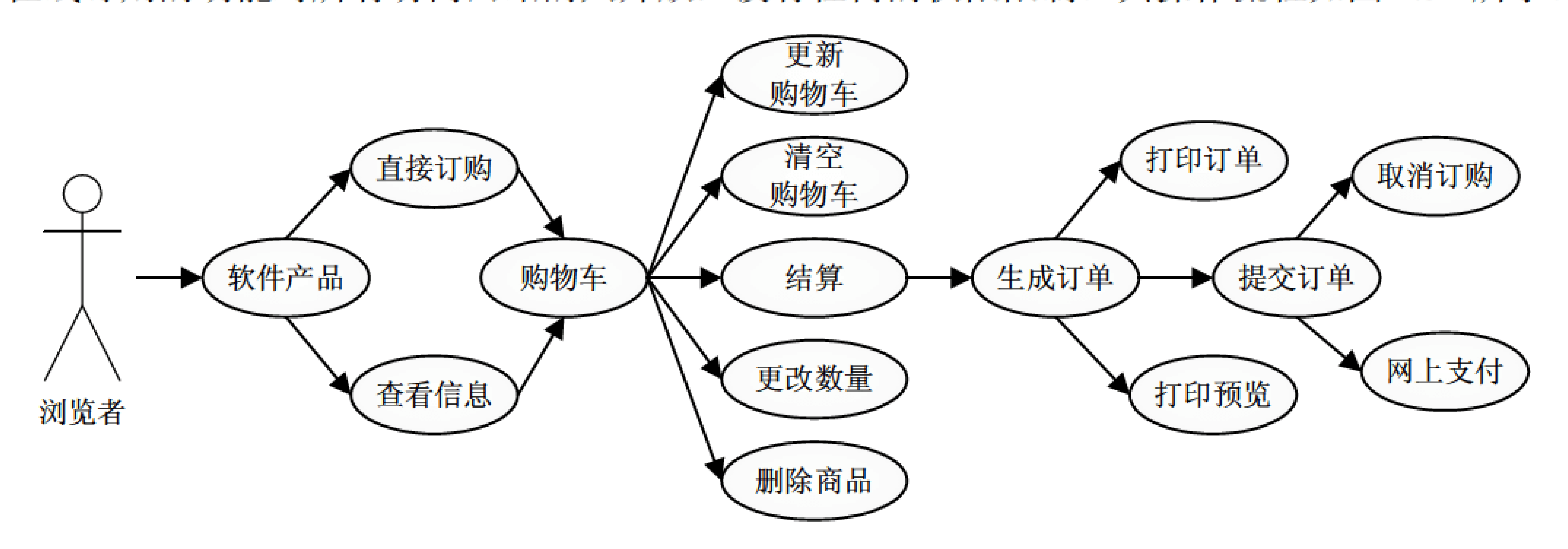


图 2.32　在线订购模块的操作流程

2.10.2　在线订购模块技术分析

在线订购管理模块中，不可缺少的一项内容就是对订单进行打印。下面就来讲解一下订单打印功能的实现方法。在线订购管理模块中运用的是 WebBrowser 打印方法。WebBrowser 是 IE 内置的浏览器控件，无须用户下载。其优点是客户端独立完成打印目标文档的生成，减轻服务器负荷；缺点是源文档的分析操作复杂，并且要对源文档中要打印的内容进行约束。

下面介绍 WebBrowser 控件的具体参数，如表 2.4 所示。

表 2.4　WebBrowser 控件的具体参数说明

参 数 名 称	说　　明
document.all.WebBrowser.Execwb(7,1)：	表示打印预览
document.all.WebBrowser.Execwb(6,1)：	表示打印
document.all.WebBrowser.Execwb(6,6)：	表示直接打印
document.all.WebBrowser.Execwb(8,1)：	表示页面设置
document.all.WebBrowser.Execwb(1,1)：	打开页面
document.all.WebBrowser.Execwb(2,1)：	关闭所有打开的 IE 窗口
document.all.WebBrowser.Execwb(4,1)：	保存网页
document.all.WebBrowser.Execwb(10,1)：	查看页面属性
document.all.WebBrowser.Execwb(17,1)：	全选
document.all.WebBrowser.Execwb(22,1)：	刷新
document.all.WebBrowser.Execwb(45,1)：	关闭窗体无提示

该技术的实现原理是：首先通过 onClick 事件调用一个 JavaScript 脚本，然后执行 openprintwindow()函数，将指定的变量值传递到订单打印页中（printwindow.php），最后在订单打印页中实现打印及打印预览的功能。调用 JavaScript 脚本和执行 openprintwindow()函数在 shopping_dd.php 页中完成。关键代



码如下：

例程09　代码位置：光盘\TM\02\bcty365\shopping_dd.php

```
<script language="javascript">
     function openprintwindow(x,y,z){                                    //定义一个函数，获取传递的参数
          //通过 window 对象中的 open 方法打开一个新窗口，并设置其属性
❶    window.open("printwindow.php?ddno="+x+"&pv="+z,"newframe","top=200,left=200,width=635,
height="+(230+20*y)+",menubar=no,location=no,toolbar=no,scrollbars=no,status=no");
       }
     </script>
<!-- ---------通过 onclick 事件调用 JavaScript 脚本，传递参数值。当参数 z 的值为 p 时执行打印功能---------- -->
<td width="75"> <img src="images/bg_14(11).jpg" width="69" height="20"
❷    onclick="javascript:openprintwindow('<?php echo base64_decode($_GET["ddno"]);?>','<?php echo
$gnum;?>','p')" style="cursor:hand"/></td>
<!-- ------通过 onclick 事件调用 JavaScript 脚本，传递参数值。当参数 z 的值为 v 时执行打印预览功能--------- -->
<td width="90"><img src="images/bg_14(12).jpg" width="69" height="20"  onclick="javascript:openprintwindow
('<?php echo base64_decode($_GET["ddno"]);?>','<?php echo $gnum;?>','v')" style="cursor:hand"/></td>
```

代码贴士

❶ open()：打开一个窗口，可以设置其 URL、名称、大小、按钮以及其他属性。该方法的基本语法结构如下：

```
     window.open(url,name,features,replace)
```

其中，url 是要在新窗口中打开文档的 URL 地址；name 是要打开窗口的名字，用 HTML 链接的 target 属性进行定位时会有用；features 是一个用逗号分隔的字符串，列举窗口的特征；replace 是一个可选的 Boolean 值，指出是否允许 URL 替换窗口的内容。

❷ onclick()：表明某元素被鼠标单击触发的事件。

订单的打印和打印预览的功能在 printwindow.php 页中完成，首先编写一个实现打印预览功能的 JavaScript 脚本，然后建立 HTML 的 object 标签，调用 WebBrowser 控件，最后获取变量传递的值，当变量的值为 p 时执行打印功能；当变量的值为 v 时执行打印预览的功能。printwindow.php 页的关键代码如下：

例程10　代码位置：光盘\TM\02\bcty365\printwindow.php

```
<script>
     function printview(){                                          //定义一个函数
❶         document.all.WebBrowser1.ExecWB(7,1) ;                   //执行 WebBrowser 控件，实现打印预览
          window.close();                                          //关闭窗口
     }
</script>
<!-- ---------------------------------建立 HTML 的 object 标签，调用 WebBrowser 控件-------------------------- -->
<object ID='WebBrowser1' WIDTH=0 HEIGHT=0
CLASSID='CLSID:8856F961-340A-11D0-A96B-00C04FD705A2'></object>
<!--- ----------------------------------在 body 中调用 onload 事件，执行打印操作----------------------------------- -->
❷    <body topmargin="0" leftmargin="0" bottommargin="0" onLoad="
<?php
     if($_GET["pv"]=="p"){                                          //判断当变量值为 p 时，执行打印操作
?>
❸    window.print();
<?php
```

```
    }elseif($_GET["pv"]=="v"){//判断当变量值为 v 时，执行打印预览操作
?>
    printview()
<?php }    ?>
">
```

代码贴士

❶ document.all.WebBrowser1.ExecWB(7,1)：WebBrowser 控件中的参数，表示打印预览。

❷ onLoad：表明某对象已载入窗口。

❸ print()：window 对象中的一个方法，实现打印的功能。

订单打印操作的运行结果如图 2.33 所示。



图 2.33　订单打印操作的运行结果

2.10.3　购物车的实现过程

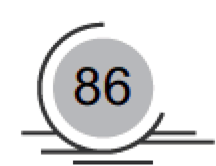
购物车使用的数据表：tb_bccd

购物车的功能是临时储存用户选购的商品，用户可以对购物车中的商品进行添加、修改、删除和更新操作，也可以选择进行结算。其运行的结果如图 2.34 所示。



图 2.34　购物车的运行结果图

购物车功能实现的第一步是为想要购买产品的用户分配一辆购物车，使其能够记录自己已经选购的产品。其工作的原理与超市中顾客使用购物车进行购物是相同的，只是这里使用的不是真正意义上的购物车，而是两个 session 变量，一个存储用户选购商品的 ID（$goodsid），另一个存储用户选购该商品的数量（$goodsnum）。如果用户在一次购物中选购多种不同类的商品，则使用@对不同类商品的不同 ID 和数量进行分割。例如，用户选购的不同类商品 id 为 1、2、3，则 session 变量$goodsid 中存储的值为“1@2@3@”；其中同一种商品不能购买两次，如果想要购买多个同种产品，可以在购物车中更改购买商品的数量。在本案例中，购物车的分配功能通过 shopping_cart_first.php 文件来完成，该文件的代码如下：



例程11　代码位置：光盘\TM\02\bcty365\shopping_cart_first.php

```
❶  <?php   session_start();                                //初始化 session 变量
❷   session_register("goodsid");                            //创建一个 session 变量
❸   session_register("goodsnum");                           //创建一个 session 变量
if($_SESSION["goodsid"]=="" && $_SESSION["goodsnum"]==""){  //判断 session 变量中的值是否为空
    $_SESSION["goodsid"]=$_GET["id"]."@";                   //如果为空，则将商品的 ID 赋给变量
    $_SESSION["goodsnum"]="1@";                             //将商品数量设置为“1@”
}else{
❹       $array=explode("@",$_SESSION["goodsid"]);           //如果不为空，则使用@分割不同的商品 ID
    //判断如果获取的 ID 在 session 变量中已经存在，则提示该商品已经被放入购物车
❺   if(in_array($_GET["id"],$array)){
        echo "<script>alert('该编程词典已经被放入购物车！');history.back();</script>";
        exit;
    }
    $_SESSION["goodsid"].=$_GET["id"]."@";                  //为 session 变量赋值
    $_SESSION["goodsnum"].="1@";                            //为 session 变量赋值
}
//将商品放入购物车中，并跳转到购物车页
echo "<script>window.location.href='shopping_cart.php';</script>";
?>
```

代码贴士

❶ session_start()：初始化 session 变量。

❷ session_register()：创建一个 session 变量 goodsid，存储商品 ID。

❸ session_register()：创建一个 session 变量 goodsnum，存储购买商品数量。

❹ explode()：将字符串依指定的字符串或字符进行分割。返回由字符串组成的数组，每个元素都是 string 的一个子串，它们被字符串 separator 作为边界点分隔出来。如果设置了 limit 参数，则返回的数组包含最多 limit 个元素，而最后那个元素将包含 string 的剩余部分；如果 separator 为空字符串（""），explode()函数将返回 false；如果 separator 所包含的值在 string 中找不到，那么 explode()函数将返回包含 string 单个元素的数组；如果参数 limit 是负数，则返回除了最后的-limit 个元素外的所有元素。

❺ in_array ()：在指定的数组中搜索某个值，如果找到则返回 true，否则返回 false。

在实现购物车的分配和添加商品的功能后，接下来要做的就是查看购物车中的商品，即实现购物车中商品展示的功能。在购物车的商品展示中，可以实现很多的操作，如清空购物车、删除购买商品、更改购买商品数量、继续购物和结算。

购物车商品展示的功能主要通过 shopping_cart.php 文件来完成，首先从 session 变量中读取商品的 ID 和数量，然后根据商品的 ID 循环输出购物车中的商品，最后以商品 ID 为标识符设置不同的超链接，执行删除商品或者更改购买商品数量等操作。代码如下：

例程12　代码位置：光盘\TM\02\bcty365\shopping_cart.php

```
<table width="720" height="47" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="#999999">
<?php
    $array=explode("@",$_SESSION["goodsid"]);          //读取 session 变量中的商品 ID，以@进行分割
    $arraynum=explode("@",$_SESSION["goodsnum"]);//读取 session 变量中的商品数量，以@进行分割
    $markid=0;                                         //创建变量，初始值为 0
    for($i=0;$i<count($array);$i++){                   //应用 for 循环语句循环输出商品 ID 的值
        if($array[$i]!=""){                            //判断如果商品 ID 的值不为空
```

```
                $markid++;                                          //增加变量$markid 的值
            }
        }
        if($markid==0){                                             //判断如果变量$markid 的值为空，则输出下面的内容
?>
        <tr>
            <td height="22" colspan="4" bgcolor="#FFFFFF"><div align="center">对不起您的购物车中暂无商品
信息!</div></td>
        </tr>
<?php
        }else{                                                      //如果$markid 的值不为空，则执行下面的内容
            $totalprice=0;                                          //创建变量$totalprice，初始值为 0
            for($i=0;$i<count($array);$i++){                        //循环输出数组中的商品 ID 值
                if($array[$i]!=""){
                    //根据获取商品 ID 的值，从数据库中获取对应产品的信息
                    $sqlcart=mysql_query("select * from tb_bccd where id='".$array[$i]."'",$conn);
                    $infocart=mysql_fetch_array($sqlcart)
?>
        <tr>
            <form name="form<?php echo $array[$i]?>" method="post" action="changegoodsnum.php">
            <td height="22" bgcolor="#FFFFFF"> <?php echo unhtml($infocart["bccdname"]);?></td>
            <td height="22" bgcolor="#FFFFFF"><div align="center"><?php echo number_format($infocart["price"],
2);?> </div></td>
            <td height="22" bgcolor="#FFFFFF"><div align="center"><input type="text" name="goodsnum" value=
"<?php echo $arraynum["$i"];?>" class="inputcss" size="8" ><input type="hidden" name="id" value="<?php
echo $infocart["id"];?>" ></div></td>
            <td height="22" bgcolor="#FFFFFF"><div align="center"><a href="javascript:form<?php echo $array
[$i]?>.submit();" class="a1">更改数量</a> | <a href="delgoods.php?id=<?php echo $infocart
["id"];?>" class="a1">删除该项</a></div></td>
            </form>
        </tr>
<?php
                    $totalprice+=$infocart["price"]*$arraynum["$i"];
                }
            }
        }
?>
</table>
```

2.10.4　商品订单的实现过程

商品订单使用的数据表：tb_bccd、tb_dd、tb_city

在确定所要购买的商品之后，接下来要做的就是进行购物结算，填写用户购物订单，将订单保存到数据库中，并且随机生成一个订单号，作为订单的唯一标识。生成订单的运行结果如图 2.35 所示。

订单提交以后用户可以选择汇款的方式：一是选择网上支付，那么将跳转到企业指定的网上银行进行汇款，汇款的操作将在企业指定的网上银行中进行，这里不做讲解；二是选择到指定的银行向企业提供的账号中汇款。企业将在收到汇款后按照用户指定的地址和方式将产品送到。商品订单的生成和处理由 shopping_cart_getuserinfo.php 和 savebuyuser.php 文件来完成。





图 2.35　生成订单的运行结果

订单处理由 savebuyuser.php 完成，首先连接数据库，随机生成一个订单号，然后获取购物车中的商品信息，最后将商品信息和订单号存储到数据库中。代码如下：

例程13　代码位置：光盘\TM\02\bcty365\savebuyuser.php

```
<?php     session_start();                                        //初始化 session 变量
include_once("conn/conn.php");                                    //连接数据库
$ddnumber=substr(date("YmdHis"),2,8).mt_rand(100000,999999);            //随机生成订单号
$sql=mysql_query("select * from tb_city where id='".$_POST["city"]."'",$conn);  //读取数据库中的城市信息
$info=mysql_fetch_array($sql);
if($shfs=="1"){                                                   //判断用户选择的送货方式
      $yprice=$info[pt];
      $shfs="普通邮递";
}elseif($shfs=="2"){
      $yprice=$info[kd];
      $shfs="邮政特快专递 EMS";
}
$array=explode("@",$_SESSION["goodsid"]);                         //以@来分割 session 变量中存储的商品 ID
$arraynum=explode("@",$_SESSION["goodsnum"]);                     //以@来分割 session 变量中存储的商品数量
$totalprice=0;
for($i=0;$i<count($array);$i++){                                  //循环读取数组中商品的 ID
      if($array[$i]!=""){
            $sqlcart=mysql_query("select * from tb_bccd where id='".$array[$i]."'",$conn);
            $infocart=mysql_fetch_array($sqlcart);
            $totalprice+=$infocart["price"]*$arraynum["$i"];
      }
}
$totalprice=$totalprice+$yprice;                                  //获取汇款金额
//将表单中提交的数据存储到数据库中
if(mysql_query("insert into
tb_dd(ddnumber,recuser,sex,address,yb,qq,email,mtel,gtel,shfs,spc,slc,yprice,totalprice,createtime,cityid)
values('".$ddnumber."','".$_POST["recuser"]."','".$_POST["sex"]."','".$_POST["address"]."','".$_POST["yb"]."','".$
_POST["qq"]."','".$_POST["email"]."','".$_POST["mtel"]."','".$_POST["gtel"]."','".$shfs."','".$_SESSION["goodsid"]
."','".$_SESSION["goodsnum"]."','".$yprice."','".$totalprice."','".date("Y-m-d H:i:s")."','".$_POST["city"]."')",$conn)){
      session_unregister("goodsid");                              //注销 session 变量 goodsid
      session_unregister("goodsnum");                             //注销 session 变量 goodsnum
      echo "<script>window.location.href='shopping_dd.php?ddno=".base64_encode($ddnumber)."';</script>";
}else{
```

```
        echo "<script>alert('订单信息保存失败，请重试!');</script>";
}
?>
```

2.10.5 单元测试

在在线订购模块的测试过程中，当执行删除购物车中某个商品的操作时，发现该项操作没能正确执行，指定的商品没有被删除。首先，从该功能实现的思路入手分析出错原因，并且仔细核对其中使用的变量以及函数是否正确。

删除商品的操作是根据商品的 ID 来进行处理的，当单击“删除”超链接时，将指定商品的 ID 通过变量$ides 传入到执行删除商品操作的文件 delgoods.php 中，然后在该文件中获取变量$ides 传入的 ID 值，并且根据 ID 值完成删除购物车中指定商品的操作。在查看该功能实现的过程中发现，定义的变量$ides 与 delgoods.php 文件中$_GET["id"]获取的变量值不一致，从而导致删除商品的操作没有正确执行，代码的出错位置如图 2.36 所示。

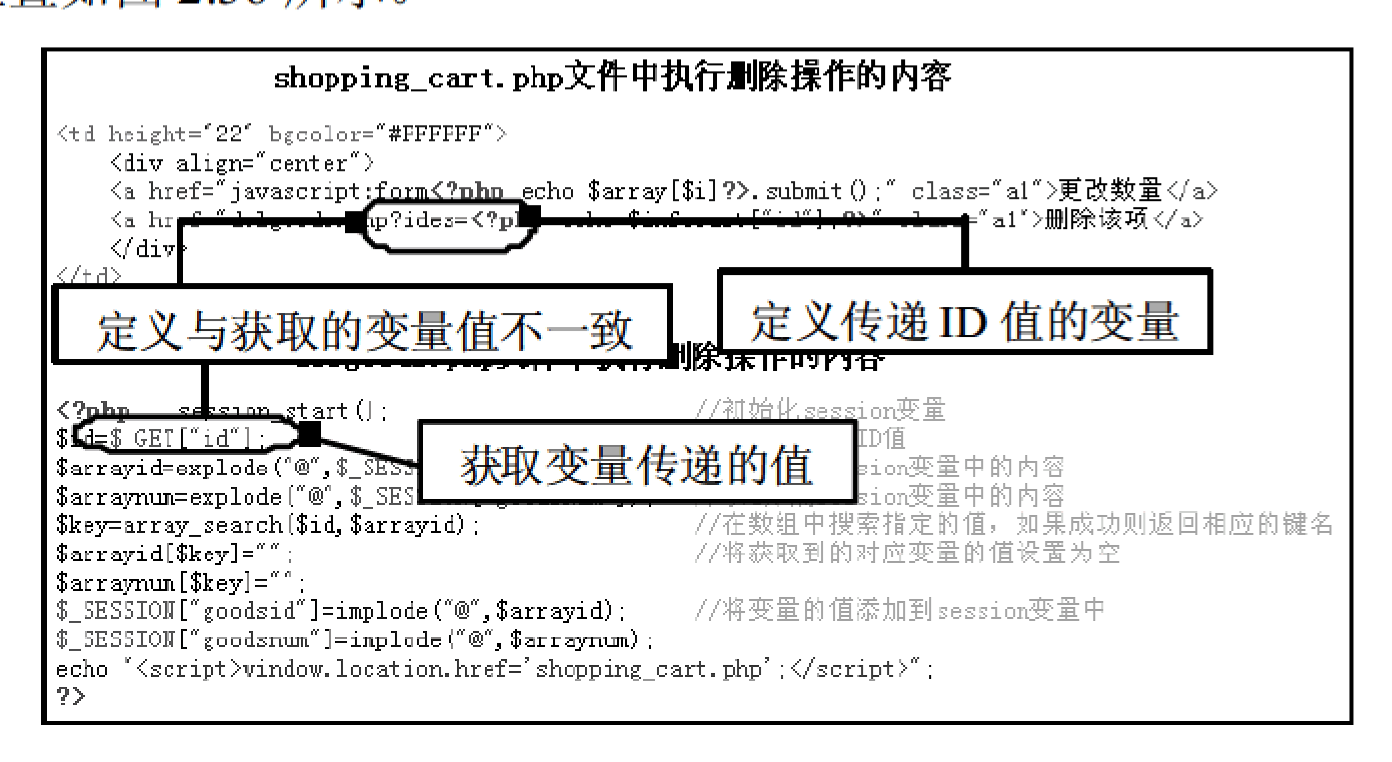


图 2.36　代码中的错误显示

查找出错误原因，将定义的变量与获取的变量值统一使用 id，保存文件，重新运行程序，删除操作可以正常运行。

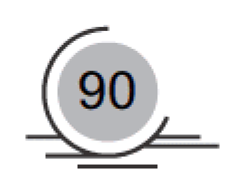
2.11 社区论坛模块设计

社区论坛模块为网站的浏览者提供一个交流的平台，以此来扩大网站的影响力，汇聚更多的人气，宣传企业形象，推广企业产品。

2.11.1 社区论坛模块概述

社区论坛模块为浏览者、会员、客户和企业之间提供一个大的交流平台，根据身份的不同，分别给予不同的操作权限。社区论坛模块的操作流程如图 2.37 所示。

在本论坛中，浏览者只能够查看帖子；注册会员既可以查看帖子，也可以发布和回复帖子；管理员则具有发布、回复、查看和删除的权限。





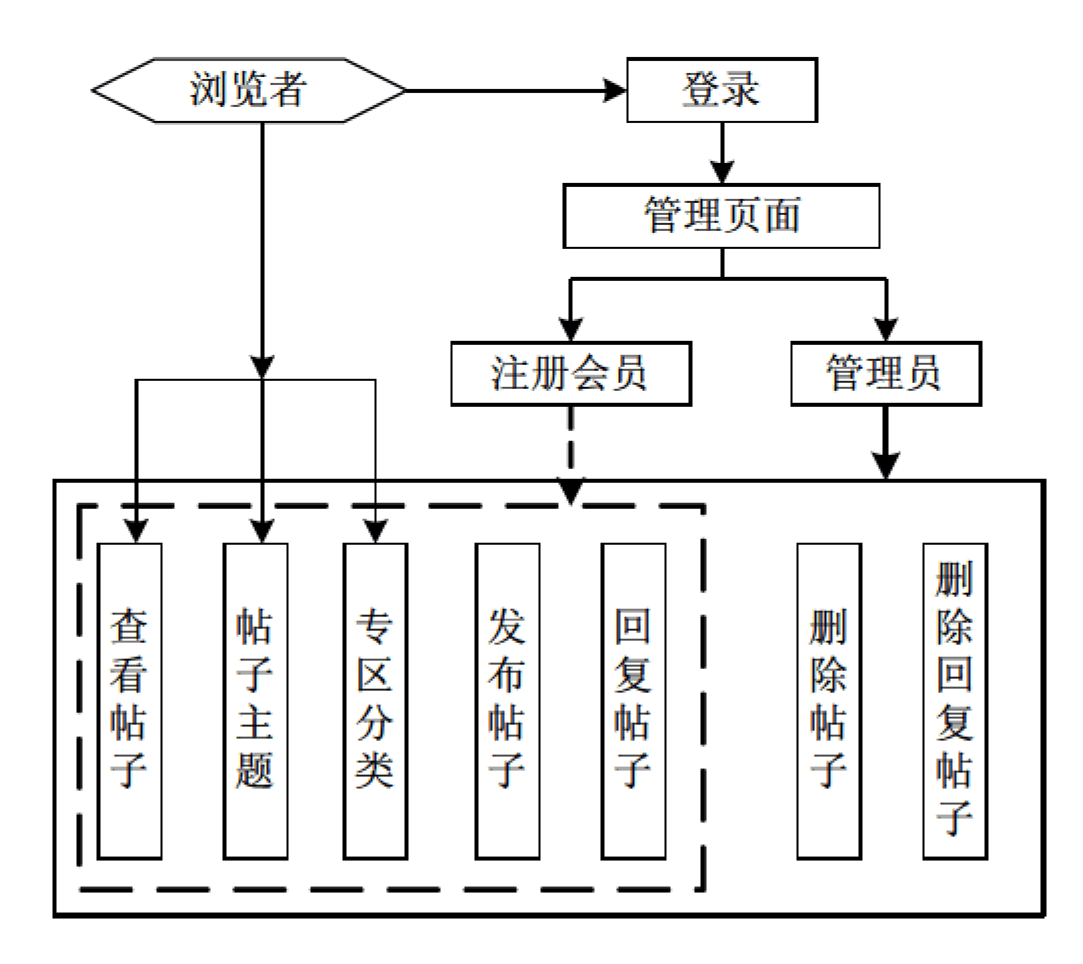


图 2.37　社区论坛流程图

2.11.2　社区论坛模块技术分析

在社区论坛模块的实现过程中，通过 JavaScript 脚本和下拉列表框的结合实现一个不同版块之间快速跳转的功能，从而能够更加灵活、方便地实现不同版块之间的跳转。

下面分析该技术是如何实现的。该技术的实现综合 3 个方面的内容，以一个下拉列表框为主，通过 PHP 语句从数据库中读取数据作为下拉列表框的值，应用 onchange 事件来调用 JavaScript 脚本，实现不同版块之间的跳转。这里以 bbs_top.php 文件中的快速跳转功能为例进行分析。关键代码如下：

例程14　代码位置：光盘\TM\02\bcty365\bbs_top.php

```
<!--创建一个下拉列表框，指定名称为 select_type，并且设置其属性，通过 onchange 事件来调用 JavaScript 脚本文件，实现页面跳转-->
<select name="select_type" class="inputcss"
❶     onChange="javascript:window.location=this.options[this.selectedIndex].value;" >
<?php
//通过 PHP 语句从数据库中读取数据，使用数据的 ID 作为下拉列表框的值，使用数据的标题 title 作为下拉列表框显示的内容
      $sql=mysql_query("select * from tb_type_small order by createtime desc",$conn);
      $info=mysql_fetch_array($sql);
      if($info==""){
            echo "<option>暂无讨论区</option>";
      }else{
            echo "<option>-版块快速跳转-</option>";
            do{                                        //应用 do…while 循环语句输出下拉列表框中的值
❷           echo "<option value='bbs_list.php?id=".$info[id]."'>".$info[title]."</option>";
            }
            while($info=mysql_fetch_array($sql));      //应用 do…while 循环语句结束
      }
?>
</select>
```

代码贴士

❶ onChange：某元素失去焦点，并且从用户最后一次访问以来，其值已经改变。

location：用于访问窗口的当前定位（URL），既可被读取，又可被置换，可以通过其实现某个页面的定位或者更新。

❷ <option value='… '>…</option>：下拉列表框中输出的值，以及显示的内容。

该技术实现的运行结果如图 2.38 所示，它将实现从硬件查询模块跳转到 PHP 模块。

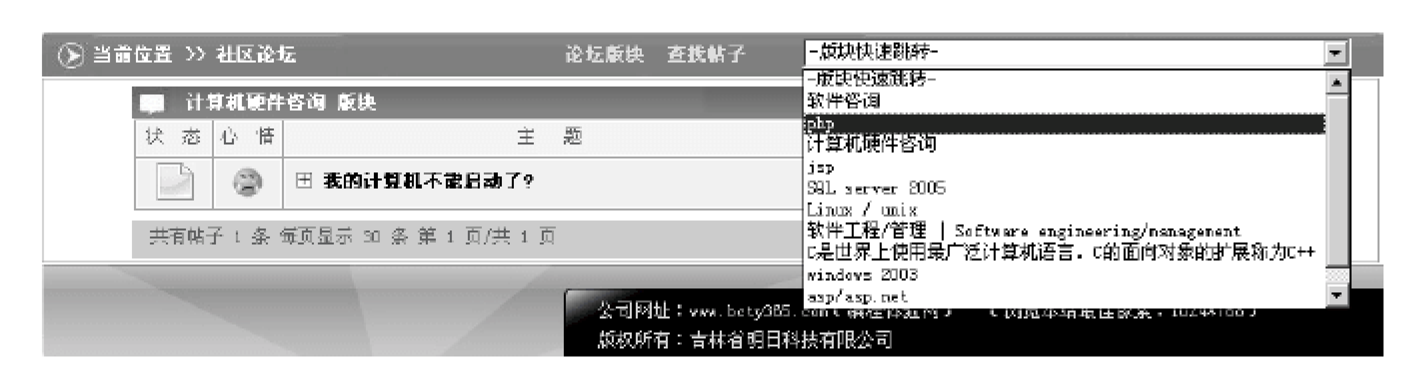


图 2.38　版块跳转功能的运行结果

2.11.3　论坛分类的实现过程

论坛分类使用的数据表：tb_bbs、tb_type_big、tb_type_small

论坛分类可以分为两类：一是论坛中大的版块分区，分为综合信息讨论区、操作系统、程序设计交流区、网管技术应用、Web 程序开发和数据库技术 6 个版块，其数据存储于 tb_type_big 数据表中。二是对应不同的版块中不同语言和技术的分类，分为 11 种，其数据存储于 tb_type_small 表中。论坛分类的运行结果如图 2.39 所示。

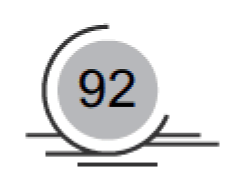


图 2.39　论坛分类的运行结果

论坛分类的实现原理很简单，首先从 tb_type_big 表中读取 6 个版块中的数据，进行循环输出，然后在版块中嵌套循环，用于输出不用语言的分类数据。该功能主要通过 bbs_index.php 文件来完成，bbs_index.php 文件的程序代码如下：

例程15　代码位置：光盘\TM\02\bcty365\bbs_index.php

```
<?php     include_once("top.php");                                  //调用网站头文件
          include_once("bbs_top.php");                              //调用社区论坛的头文件
?>
<?php
//循环输出数据表 tb_type_big 中的 6 个版块数据
$sql=mysql_query("select * from tb_type_big order by createtime desc",$conn);
$info=mysql_fetch_array($sql);
if($info==false){                                                   //如果返回值为 false，则执行下面的内容
?>
……省略了部分代码
<?php
     }else{                                                         //如果返回值为 true，则执行 do…while 循环语句
          /* 外部嵌套循环，输出论坛中的版块分类数据 */
❶         do{
?>
……省略了部分代码
<table width="750" border="0" align="center" cellpadding="0" cellspacing="1" bordercolor="#FFFFFF"
```



```
bgcolor="#6EBEC7">
<?php
    //循环输出 tb_type_small 表中的不同语言和技术的分类数据
    $sql1=mysql_query("select * from tb_type_small where bigtypeid='".$info["id"]."'",$conn);
    $info1=mysql_fetch_array($sql1);
    if($info1==false){                          //判断如果返回值为 false，则执行下面的内容
?>
……省略了部分代码
<?php
    }else{        //如果返回值为 true，则执行下面的内容，输出该版块中对应语言和技术的帖子详细信息
?>
……省略了部分代码
❷  <?php   /* 内部嵌套循环，输出不同语言和技术的分类 */
            do{          ?>
        <tr>
            <td height="30"><font color="#666666">创建时间：<?php echo $info1["createtime"];?>
</font></td>
        </tr>
……省略了部分代码
      <?php
            }while($info1=mysql_fetch_array($sql1));
        /* 内部嵌套循环结束 */
    }
    ?>
</table>
  <?php
        }while($info=mysql_fetch_array($sql));
      /* 外部嵌套循环结束，对版块中的大类进行输出 */
}
 ?>
<?php    include_once("bottom.php");       ?>
```

代码贴士

❶ do…while 循环语句，对论坛中大的版块分类进行循环输出。

❷ do…while 循环语句，对论坛中一个版块的不同语言和技术进行循环输出。

注意　在应用 do…while 循环语句时，while 后的分号不能省略。

2.11.4　论坛帖子浏览的实现过程

论坛帖子浏览使用的数据表：tb_bbs、tb_user、tb_reply

论坛帖子浏览主要输出指定帖子的详细信息，包括发帖人、用户级别和注册的时间，以及帖子的主题、内容和发帖时间，包括上传的图片。本模块是用户权限使用体现的最明显地方，可以分为 3 种情况：第一以浏览者进行登录，只能是浏览帖子的内容，没有其他权限；第二以会员进行登录，可以对帖子进行回复，发表自己的看法；第三以管理的身份进行登录，不但可以回复帖子，而且可以对任何人发布和回复的帖子进行删除和顶帖的操作。下面就来看一下以管理员身份进行登录时都具备哪些权限，运行结果如图 2.40 所示。



图 2.40　管理员浏览帖子的结果图

论坛帖子浏览的功能通过 bbs_looks.php 文件完成，首先根据传递的 ID 值读取指定的帖子数据，然后判断登录用户的类型，最后根据用户不同的类型执行不同的操作。代码如下：

例程16　代码位置：光盘\TM\02\bcty365\bbs_look.php

```
<?php
//根据$_GET 传递的数据获取 tb_bbs 中的数据
$sqlb=mysql_query("select * from tb_bbs where id='".$_GET["id"]."'",$conn);
$infob=mysql_fetch_array($sqlb);
//根据$_GET 传递的数据获取 tb_user 中的数据
$sql4=mysql_query("select * from tb_user where id='".$infob["userid"]."'",$conn);
$info4=mysql_fetch_array($sql4);
?>
…                                                         //省略了部分 HTML 代码
<table width="180" border="0" align="center" cellpadding="0" cellspacing="0">
    <tr>
        <td height="22">用户级别：
            <?php
                //根据用户信息表 tb_user 中字段 usertype 的值判断该用户的类型
                //如果值为 1 则是管理员，值为 2 则是后台管理员，值为 0 则是普通会员
❶               if($info4["usertype"]=="1") echo "管理员";else echo "普通会员";
            ?>
        </td>
    </tr>
</table>
…                                                     //省略了部分 HTML 代码
<table width="500" height="200" border="0" align="center" cellpadding="0" cellspacing="0">
    <tr>
        <td width="35" height="20"><div align="center"><img src="images/lt_15(11).jpg" width="25" height=
"25"></div></td>
       <td><?php echo $infob["createtime"];?></td>
    </tr>
    <tr>
        <td height="150" colspan="2">
            <?php
            //判断 tb_bbs 表中的字段 photo 是否为空，为空则执行下面的内容
❷               if($infob[photo]!=""){
```

```
                    $photos=substr($infob[photo],2,70);          //获取图片在服务器中的存储路径
                    echo (stripslashes($infob["content"])); //输出帖子的内容
                    echo "<img src=\"$photos\">";                //根据获取的图片路径，输出服务器中的图片
                }else{                              //如果 tb_bbs 表中的图片字段 photo 为空，则执行下面的内容
❸                   echo (stripslashes($infob["content"])); //只输出帖子的内容
                }
            ?>
        </td>
    </tr>
</table>
<table width="530" height="20" border="0" align="center" cellpadding="0" cellspacing="0">
    <tr>
        <td width="239" height="30"> </td>
        <td width="291"><img src="images/lt_15(5).jpg" width="72" height="23" style="cursor:hand" onclick="
            <?php
                //如果$_SESSION["unc"]的值为空，则不可以进行顶帖子的操作
                if($_SESSION["unc"]==""){
                    echo "javascript:alert('请先登录本站，然后进行此操作！');window.location.href='index.php';";
                }else{
                    //否则将判断当前用户的类型，如果是管理员则可以顶帖
                    $sqlu=mysql_query("select usertype from tb_user where
usernc='".$_SESSION["unc"]."'",$conn);
                    $infou=mysql_fetch_array($sqlu);
❹                   if($infou["usertype"]==1){          //如果用户的类型为 1，则有顶帖的权限
                        echo "javascript:window.location.href='settop.php?id=".$infob["id"]."'";
                    }else{                              //否则不具备该权限
                        echo "javascript:alert('对不起，您不具备该操作权限！');";
                    }
                }
            ?>
            "/>  
❺           <?php
                //判断当前用户是否具有删除帖子的权限
                if($_SESSION["unc"]!=""){
                    //条件为用户不能为空，并且是管理员，才具备删除帖子的权限
                    $sqlu=mysql_query("select usertype from tb_user where
usernc='".$_SESSION["unc"]."'",$conn);
                $infou=mysql_fetch_array($sqlu);
                if($infou["usertype"]==1){
            ?>
            <img src="images/lt_15(10).jpg" onclick="javascript:if(window.confirm('您确定删除该帖吗？
')==true){window.location.href='bbs_delete.php?id=<?php echo $infob["id"]?>';}" style="cursor:hand"/>
            <?php
                    }
                }
            ?>
        </td>
    </tr>
</table>
```

代码贴士

❶ $info4["usertype"]：判断用户的类型，如果值为 1 是管理员，否则为普通会员。
❷ $infob[photo]：判断发布的帖子中是否含有图片，如果有则输出，没有则不输出。
❸ stripslashes()：将应用 addcslashes()函数处理后的字符串返回原样。
❹ 判断登录用户是否具有顶帖的权限。
❺ 判断登录用户是否具有删除帖子的权限。

说明 上面给出的是该文件的部分代码，主要讲解了该功能的实现方法，完整的代码可以参考本书光盘中的 TM\02\bcty365\bbs_lookbbs.php 文件。

2.11.5 论坛帖子发布的实现过程

论坛帖子发布使用的数据表：tb_bbs、tb_user

论坛帖子发布通过两个文件来完成：一个是帖子发布信息的提交页 bbs_pubs.php；另一个是对提交的数据进行处理的 retrieve.php 文件。该功能实现的运行结果如图 2.41 所示。

在发布信息的提交页中，显示当前用户的个人信息，设置添加数据表单元素，其中表单元素的设计如表 2.5 所示。



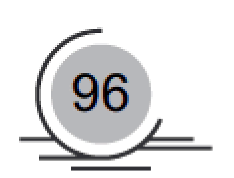
图 2.41 帖子发布模块的运行结果

表 2.5 发布信息页中使用的表单元素

名　称	元 素 类 型	重 要 属 性	含　义
form_bbs	form	method="post" action="retrieve.php" enctype="multipart/form-data"	发帖表单

bbs_type	select	class="inputcss" style="background-color:#6EBEC7">
		<?php
		$sql=mysql_query("select * from tb_type_small order by createtime desc",$conn);
		$info=mysql_fetch_array($sql);
		if($info==false){
		echo "<option>暂无讨论区</option>";
		}else{
		do{
		?>
		<option value="<?php echo $info[id] ;?>"<?php if($_GET[id]==$info[id]){echo "selected=\"selected\"";}?>><?php echo $info[title];?></option>
		<?php }
		while($info=mysql_fetch_array($sql));
		}?>
			选择发表帖子的语言或者技术的类别

bbs_title	text	class="inputcss" style="background-color:#6EBEC7">	帖子标题
bbs_head	radio	value="<?php echo("images/bbsface/face".($i-1).".gif");?>"	表情图



续表

名　称	元 素 类 型	重 要 属 性	含　义
bbs_photo	file	id="bbs_photo" class="inputcss" style="background-color:#6EBEC7" />	上传图片
content1	textarea	id="content1" class="inputcss" style="background-color:#6EBEC7">	帖子内容
Submit	submit	value="提交"	提交表单

在 retrieve.php 页中对表单提交的数据进行处理，将数据存储到 tb_bbs 表中，并且更新用户信息表 tb_user 中 pubtimes 字段的值，其中还应用了图片上传技术，将图片上传到服务器中指定的文件夹下。retrieve.php 文件的代码如下：

例程17　代码位置：光盘\TM\02\bcty365\retrieve.php

```
<?php    session_start();                                     //初始化 session 变量
$title=$_POST[bbs_title];                                      //获取帖子的标题
$content=$_POST[content1];                                     //获取帖子的内容
/*判断提交的帖子主题和帖子内容是否为空*/
if($title==""){
     echo "<script>alert('请输入帖子主题!');history.back();</script>";
     exit; }
if($content==""){
     echo "<script>alert('请输入帖子内容!');history.back();</script>";
     exit; }
/*********/
include_once("conn/conn.php");                                 //连接数据库
//根据$_SESSION["unc"]的值读取数据库中用户的信息
$sql=mysql_query("select * from tb_user where usernc='".$_SESSION["unc"]."'",$conn);
$info=mysql_fetch_array($sql);                                 //检索指定条件的数据信息
$userid=$info[id];                                             //获取用户 id
$typeid=$_POST[bbs_type];                                      //接收版块名称
$title=$_POST[bbs_title];                                      //接收帖子主题
$content=$_POST[content1];                                     //接收帖子内容
$head=$_POST[bbs_head];                                        //接收头像
$createtime=date("Y-m-j H:i:s");                               //获取系统当前时间
$lastreplytime=$createtime;                                    //将当前时间赋给变量
$readtimes=0;
$link=date("YmjHis");
❶    if($_FILES['bbs_photo']["name"]==true){      //上传图片，判断文件是否存在，如果存在则执行下面的内容
❷    $photo_name=strtolower(stristr($_FILES["bbs_photo"]["name"],".")); //获取图片后缀名，将字符转换成小写
     if($photo_name!=".gif" & $photo_name!=".jpg" & $photo_name!=".jpeg" ){ //判断图片的格式是否符合要求
          echo "<script>alert('您上传的图片格式不正确!');history.back();</script>";
     }else{
❸         $paths1=$link.mt_rand(1000000,9999999).$photo_name;           //创建图片的名称
          $photos="./upfile/".$paths1;                                   //创建图片的存储路径
❹              move_uploaded_file($_FILES['bbs_photo']["tmp_name"],$photos);    //将图片存储到指定的文件夹下
          //向数据库添加数据
          if(mysql_query("insert into tb_bbs(userid,typeid,title,content,createtime,lastreplytime,head,readtimes,top,photo) values('".$userid."','".$typeid."','".$title."','".$content."','".$createtime."','".$lastreplytime."','".$head."', '".$readtimes."','0','$photos')",$conn)){
```

```
                mysql_query("update tb_user set pubtimes=pubtimes+1",$conn);     //更新 tb_user 中 pubtimes
字段的值
                echo "<script>alert('新帖发表成功!');history.back();</script>";
            }else{
                echo "<script>alert('新帖发表失败!');history.back();</script>";
            }
        }
}else{              //如果没有提交图片，则执行下面的内容
        if(mysql_query("insert  into  tb_bbs(userid,typeid,title,content,createtime,lastreplytime,head,readtimes,top)
values  ('".$userid."','".$typeid."','".$title."','".$content."','".$createtime."','".$lastreplytime."','".$head."',  '".$readtimes."',  '0')",
$conn)){
            mysql_query("update tb_user set pubtimes=pubtimes+1",$conn);
            echo "<script>alert('新帖发表成功!');history.back();</script>";
        }else{
            echo "<script>alert('新帖发表失败!');history.back();</script>";
        }
}
?>
```

代码贴士

❶ $_FILES['bbs_photo']["name"]：$_FILES[]全局变量，获取表单提交文件的原始名称。

❷ strtolower()：将指定的字符转换为小写字母。

stristr()：获取指定字符串（A）在另一个字符串（B）中首次出现的位置到（B）字符串末尾的所有字符串。该函数如果执行成功则返回剩余的字符串，否则将返回 false。

❸ mt_rand()：生成一个随机数，用于上传文件的名称。

❹ move_uploaded_file()：将指定的文件上传到指定的文件夹下。

2.11.6　论坛帖子回复的实现过程

论坛帖子回复使用的数据表：tb_bbs、tb_user、tb_reply

回复论坛中的帖子，必须是以会员或者管理员的身份进行登录，否则不能进行帖子的回复操作，其运行结果如图 2.42 所示。

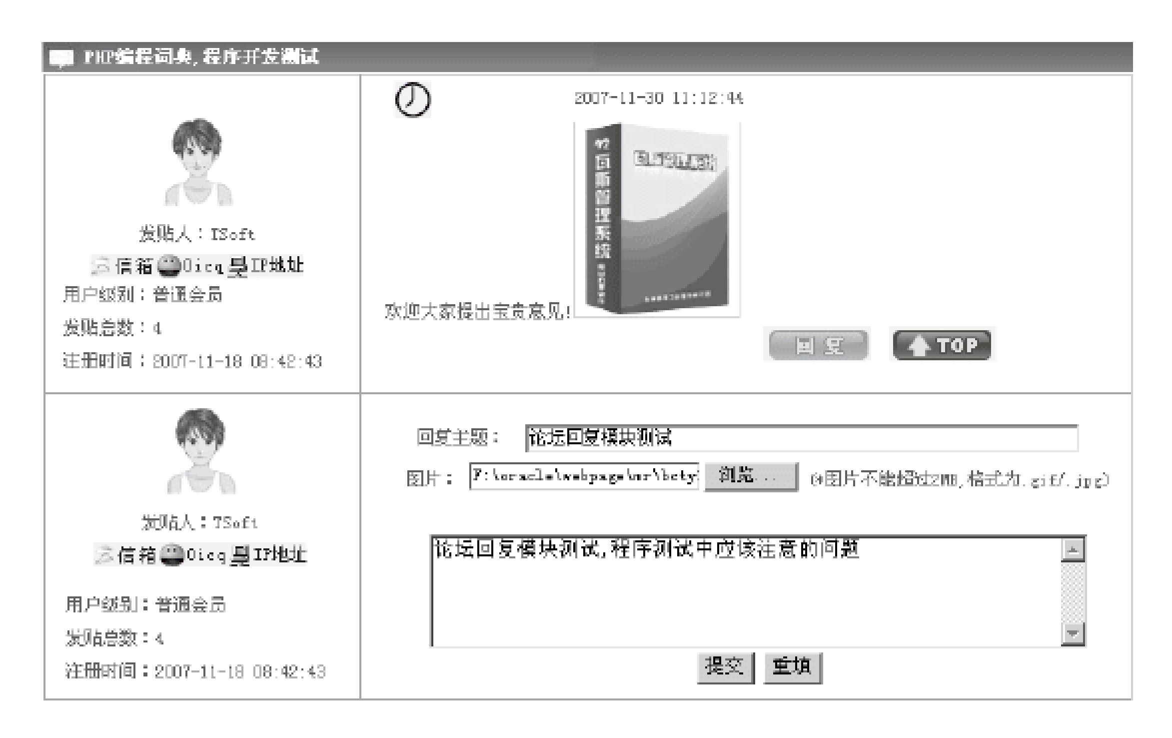
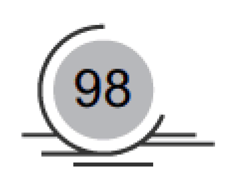


图 2.42　论坛帖子回复的运行结果

论坛帖子回复功能的实现主要通过 bbs_looks.php 和 savereply.php 两个文件。其中应用 JavaScript 脚本对回复帖子的文本框进行输出和隐藏的控制。在 bbs_looks.php 文件中，帖子回复使用的表单元素



如表 2.6 所示。

表 2.6　论坛帖子回复中的重要表单元素

名　称	元 素 类 型	重 要 属 性	含　义
form_reply	form	method="post" action="savereply.php" enctype="multipart/form-data">	回复表单
reply_title	text	class="inputcss" id="reply_title"	回复帖子主题
bbsid	hidden	value="<?php echo $infob["id"];?>"	对应帖子的 ID
bbs_head	radio	value="<?php echo("images/bbsface/face".($i-1).".gif");?>"	表情图
bbs_photo	file	id="bbs_photo" class="inputcss"	上传图片
content1	textarea	id="content1"	回复帖子内容
Submit	submit	value="提交"	提交表单

在帖子回复表单 bbs_looks.php 页中，首先判断登录用户是否具有回复的权限，然后根据提交的值展开回复表单的文本框，在文本框中输入回复的主题和内容，最后将数据提交到表单处理页 savereply.php 中。bbs_looks.php 的主要代码如下：

例程18　代码位置：光盘\TM\02\bcty365\ bbs_lookbbs.php

```
<script language="javascript">
//设计回复帖子表格的输出方式
function show_reply(){                                          //定义一个函数
      if(reply_bbs1.style.display=="")                          //判断当 display 的值为空时
            {
                  reply_bbs1.style.display="none" ;             //输出表格
                  button_show_bbs.value="回复帖子";              //显示回复帖子
            }
      else if(reply_bbs1.style.display=="none")                 //判断当 display 的值为 none 时
            {
                  reply_bbs1.style.display="" ;                 //不输出表格
                  button_show_bbs.value="关闭窗口";              //显示关闭窗口
             }
}
</script>
<!-- ------------------------------------判断登录用户是否具有回复的权限---------------------------------------- -->
<img src="images/lt_15(9).jpg" width="72" height="23" id="button_show_bbs"   style="cursor:hand" onClick="
<?php
      if($_SESSION["unc"]==""){
            echo "javascript:alert('请先登录本站，然后回复帖子!');window.location.href='index.php';";
      }else{
?>
      show_reply()
<?php      }      ?>          "/>
```

表单处理页 savereply.php 将表单提交的数据存储到指定的数据库中，其实现的方法与论坛发布中的表单处理技术是相同的，有关该技术的详细讲解请参考 2.11.5 节，这里不再赘述。

2.11.7　单元测试

在测试网上社区的论坛模块时，发现发帖和回帖时上传的图片不能够正常显示，运行结果如图 2.43

所示。分析错误原因，主要有两个方面：一是图片没有上传到指定的服务器文件夹下；二是上传成功后，没能正确地读取数据库中指定图片的路径。

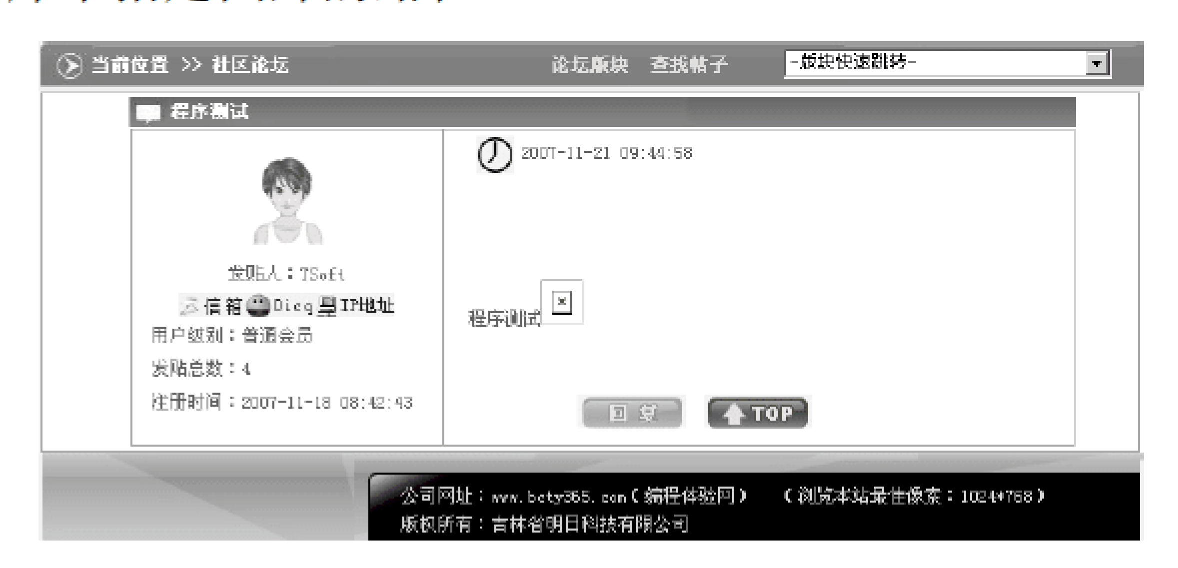


图 2.43　程序运行错误结果图

首先，检测第一种情况，没有出现问题，图片可以上传到指定的文件夹下，并且图片的路径也可以存储到指定的数据表中，从而排除了第一种情况的可能。

然后，看第二种情况，查看获取的图片路径是否正确。发现在读取数据库中图片路径的代码段中，使用了错误的字段名称，数据库中图片路径存储使用的字段名是 photo，而在程序代码段中使用的却是 photos。错误代码如下：

```
<?php
	if($infob[photo]!=""){                                   //判断是否存在图片
		$photos=substr($infob[photos],2,70);                 //获取图片存储的路径
		echo (stripslashes($infob["content"]));              //输出图片的内容
		echo "<img src=\"$photos\">";                        //输出图片
	}else{
		echo (stripslashes($infob["content"]));              //输出帖子内容
	}
?>
```

将代码段中的字段名进行修改，然后重新运行程序，图片正常显示。

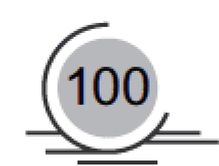
2.12　后台首页设计

作为一个完整的网上社区系统，要想能够及时地对网站进行管理和维护，必须具有一个强大的后台管理系统，对网上社区系统中的数据进行更新和维护。

2.12.1　后台首页概述

网上社区系统的后台管理采用的是一种简单的框架结构，通过 switch 语句来实现。其包括如下具体内容。

- ☑ 软件试用管理：包括软件试用产品的添加和删除。
- ☑ 编程词典管理：包括编程词典版本的添加、删除和编程词典内容的添加和删除。
- ☑ 在线订购管理：主要用于管理用户提交的订单。
- ☑ 软件升级管理：包括升级包的添加、删除和序列号的添加和删除。



☑　站内公告管理：主要用于添加和删除站内公告。

☑　技术支持管理：主要用于添加常见问题和删除常见问题，以及对客户反馈信息进行管理。

本案例中提供的后台首页如图 2.44 所示。该页面在本书光盘中的路径为\TM\02\bcty365\admin\index.php。



图 2.44　BCTY365 网上社区系统后台首页

2.12.2　后台首页技术分析

网上社区后台首页的设计主要应用 switch 语句和 include 包含语句，其实现的原理是：应用 switch 语句，根据超链接中传递的变量值进行判断，根据不同的变量值应用 include 调用不同的子文件。该技术的实现流程如图 2.45 所示。



图 2.45　网上社区后台首页设计流程

为了能够更好地理解这个技术，先来了解一下 switch 语句。该语句的格式如下：

```
switch( expr ){                          //expr 条件为变量名称
    case expr1:                          //case 后的 expr1 为变量的值
        statement1;                      //冒号（:）后的是符合该条件时要执行的部分
    break ;                              //应用 break 来跳离循环体
   case expr2 :
        statement2 ;
    break ;
    default:
        statementN;
    break;
}
```

参数 expr 是表达式的值，即 switch 语句的条件变量的名称；参数 expr1 放置于 case 语句之后，是要与条件变量 expr 进行匹配的值中的一个；statement1 是在参数 expr1 的值与条件变量 expr 的值相匹配时执行的代码；break 语句实现终止语句的执行，即当语句在执行过程中，遇到 break 就停止执行，跳出循环体；default 是 case 的一个特例，匹配了任何其他 case 都不匹配的情况，并且是最后一条 case 语句。

通过 switch 和 include 语句来实现后台管理功能的设计是一个很好的方法，不但实现过程简单，而且操作也非常灵活。关键代码如下：

例程19　代码位置：光盘\TM\02\bcty365\ admin\wzdh.php

```
<?php
switch($htgl){                                          //根据变量提交的不同值
    case "添加编程词典版本":                            //判断与变量提交的值是否相同
        include("addbb.php");                           //如果值相同，则调用指定的文件
    break;                                              //并且跳出本次循环
  case "编辑编程词典版本":
  include("editbd.php");
    break;
      …                                                 //部分代码省略
  case "":                                              //当变量的值为空时
  include("edittell.php");                              //调用该文件
    break;
}
?>
```

2.12.3　后台首页的实现过程

*后台首页使用的数据表：*tb_bb

在后台首页的设计过程中，以 switch 循环语句为基础，架设整个后台管理功能的框架结构；充分发挥 include 包含语句的作用，调用不同的文件执行不同的管理操作；应用 JavaScript 脚本来控制栏目列表的输出和隐藏。

控制栏目列表的输出和隐藏在 menu.php 文件中进行，首先定义一个函数 change()用于控制表格的输出和隐藏，然后在表格中应用 onclick 事件传递不同的值到自定义函数 change()，最后根据不同的值显示不同的内容。关键代码如下：

例程20　代码位置：光盘\TM\02\bcty365\admin\menu.php

```
<script language="javascript">
//通过脚本语言控制文本框的伸展和收缩
function change(x,y){                                   //定义一个函数
     if(x.style.display=="none"){                       //判断当样式的值为 none 时
         x.style.display="";                            //输出样式的值为空
     }
     else if(x.style.display==""){                      //判断当样式的值为空时
         x.style.display="none";                        //输出样式的值为 none
         y.background="images/bg_16_11.jpg";            //输出背景图片
           }
     }
```

```
</script>
<table width="175" height="28" border="0" align="center" cellpadding="0" cellspacing="4"
onclick="change(tz1,img_tz1)" style="cursor:hand">
  <tr>
    <td background="images/bg_16_11.jpg" id="img_tz1" class="a4"><div align="left"><img
src="images/bg_16_21.jpg"> 编程词典管理</div></td>
  </tr>
</table>
<table name="tz1" id="tz1" width="170" height="40" border="0" align="center" cellpadding="0" cellspacing="0"
<?php
     //根据变量的值选择执行的内容，当变量的值不为真时，隐藏该表格
     if(!($_GET[htgl]=="添加编程词典版本" ||$_GET[htgl]=="编辑编程词典版本" ||$_GET[htgl]=="添加编程词典"
||$_GET[htgl]=="编辑编程词典" )){
?>
     style="display:none"
<?php     }   ?>
>
   <tr>
    <td width="40" height="24" background="images/bg_16_16.jpg"> </td>
    <td width="114" background="images/bg_16_16.jpg"><div align="left"><a href="default.php?htgl=添加编程
词典版本">添加编程词典版本</a></div></td>
  </tr>
</table>
```

说明 这里给出的只是后台首页实现过程中的主要代码，详细代码可参考本书光盘 TM\02\bcty365\admin\文件夹下的相关文件。

2.13　编程词典管理模块设计

本模块的功能是对网站中的编程词典进行管理，包括添加编程词典版本、编辑编程词典版本、添加编程词典和编辑编程词典。

2.13.1　编程词典管理模块概述

本模块的主要功能是管理网站中在线出售的编程词典软件，实现对编程词典软件的及时更新和维护，其管理的内容主要包括添加和编辑编程词典的版本，添加和编辑编程词典的详细信息。在添加编程词典时，包括名称、版权、图片、类别、内容简介和不同版本的共同点；编辑编程词典包括版本、价格、简介、功能和服务，其中每一个编程词典软件只可以编辑一次，不可以进行重复编辑，如果要重新编辑，就必须将已经编辑过的信息删除。

2.13.2　编程词典管理模块技术分析

在编程词典管理模块中，应用到图片上传技术，通过该技术将编程词典的界面效果上传到服务器

的指定文件夹下。该技术主要通过 move_uploaded_file()函数来实现，其中还应用到 is_dir()、mkdir()函数，判断指定的文件夹是否存在和创建文件夹，还有 mt_rand()、strstr()函数和$_FILES[]全局变量。为了更好地理解和掌握图片上传处理技术，这里以编程词典模块中的 savebccd.php 文件为例进行讲解。

首先应用 is_dir()函数判断在服务器中是否存在指定的文件夹，如果不存在，则应用 mkdir()函数创建一个新的文件夹。

然后应用$_FILES[]全局变量获取图片名，应用 strstr()函数获取图片文件的后缀名，为了避免出现同名文件覆盖，这里应用系统的当前时间和 mtrand()函数获取的一个 7 位随机数字作为图片的名称。

最后确定图片在服务器中存储的路径，将图片上传到指定的文件夹下。而数据库中存储的数据是图片在服务器中的路径，当需要输出图片时，只需要获取到数据库中图片的路径即可。savebccd.php 文件的代码如下：

例程21 代码位置：光盘\TM\02\bcty365\admin\savebccd.php

```
<?php include_once("../conn/conn.php");                                   //连接数据库
$bccdname=$_POST[bccdname];                                              //获取 POST 方法提交的值
$owner=$_POST[owner];
$typeid=$_POST[typeid];
$content=$_POST[content];
$samepart=$_POST[samepart];
$addtime=date("Y-m-j H:i:s");                                            //获取当前时间
❶    if(is_dir("./bccdimages")==false){                                   //判断指定的文件是否存在
❷    mkdir("./bccdimages");                                               //如果不存在，则创建一个新的文件夹
}
$link=date("YmjHis");                                                    //获取当前时间
//为表单中提交的数据重新命名，以当前时间和随机数作为名称
//其中使用$_FILES 获取表单中真实的名称，使用 strstr()函数获取文件的后缀
❸    $path=$link.mt_rand(1000000,9999999).strstr($_FILES["imageaddress"]["name"],".");
$address="./bccdimages/".$path;                                          //定义文件上传的路径
❹    move_uploaded_file($_FILES["imageaddress"]["tmp_name"],$address);   //将文件上传到指定的文件
中
$imageaddress="./admin/bccdimages/".$path;                              //获取上传文件在服务器中的存储路径
//将表单中提交的数据存储到数据库中
$query=mysql_query("insert into tb_bccd(bccdname,owner,typeid,content,samepart,imageaddress,addtime)
values('$bccdname','$owner','$typeid','$content','$samepart','$imageaddress','$addtime')",$conn);
if($query==true){
     echo "<script>alert('编程词典添加成功！');history.back();</script>";
}else{
     echo "<script>alert('编程词典添加失败！');history.back();</script>";
}
?>
```

代码贴士

❶ is_dir()：判断指定的文件夹是否存在，如果存在则返回 true，否则返回 false。

❷ mkdir()：创建一个新的文件夹。

❸ mt_rand()：根据提供的参数 min 和 max 生成随机数，如果没有提供可选参数 min 和 max，则返回 0 到 RAND_MAX 之间的伪随机数。

strstr()：获取一个指定字符串在另一个字符串中首次出现的位置到后者末尾的子字符串。如果执行成功，则返回剩余字符串（存在相匹配的字符）；如果没有找到相匹配的字符，则返回 false。



$_FILES[]: 全局变量，获取所有上传文件的信息。该全局变量还可以获取到其他的值，其中$_FILES['imageaddress']['name']获取的是客户端机器文件的原名称；$_FILES['imageaddress']['size']获取已上传文件的大小，单位为字节；$_FILES['imageaddress']['tmp_name']获取文件被上传后在服务端存储的临时文件名；$_FILES['imageaddress']['error']获取和该文件上传相关的错误代码。

❹ move_uploaded_file(string filename,string destination)：应用 POST 方法实现文件的上传，参数 filename 指定要上传的文件地址；参数 destination 指定文件上传到服务器后的存储目录及名称。

注意 应用 POST 方法上传图片文件时，应当在上传表单的<form>标记中添加内容 "enctype="multipart/ form-data""。

2.13.3　添加编程词典的实现过程

添加编程词典使用的数据表：tb_bccd、tb_type

添加编程词典的功能是向数据库中添加编程词典的详细信息，包括编程词典的名称、版权、图片、类别、内容简介和不同版本的共同点。其运行结果如图 2.46 所示。

添加编程词典

编程词典名称：	PHP编程词典
版权所有：	吉林省明日科技有限公司
图片路径：	C:\Documents and Settings\ 浏览...
所属类别：	〖PHP〗类
内容简介：	整合当前大部分比较流行的PHP以及相关的技术，为广大的PHP爱好者提供一个好助手.
不同版本相同点：	针对不同的版本，其中涉及到不同难易程度的知识
	添加　重写

图 2.46　添加编程词典模块的运行结果

添加编程词典信息模块主要通过 addbccd.php 和 savebccd.php 文件来完成，其中在 addbccd.php 文件中主要是设计表单元素，而 savebccd.php 文件主要是对表单中提交的数据进行处理。addbccd.php 文件中使用的表单元素如表 2.7 所示。

表 2.7　添加编程词典页中使用的重要表单元素

名　称	元素类型	重要属性	含　义
form1	form	method="post" action="savebccd.php" onSubmit="return chkinput(this)" enctype="multipart/form-data">	编程词典表单
bccdname	text	size="25" class="txt_grey"	编程词典名称
owner	text	size="25" class="txt_grey"	版权所有者
imageaddress	file	size="25" class="txt_grey"	界面图片
typeid	select	<?php　include_once("../conn/conn.php"); $sql=mysql_query("select * from tb_type order by createtime desc",$conn); $info=mysql_fetch_array($sql); if($info==false){ echo "<option >暂无类别</option>";	选择编程词典的版本

续表

名　称	元素类型	重要属性	含　义

typeid	select	}else{
		do{
		echo "<option value=".$info[id].">".$info[typename]."</option>"; }
		while($info=mysql_fetch_array($sql));}　　?>
			选择编程词典的版本

content	textarea	rows="10" cols="65" class="textarea"	内容简介
samepart	textarea	rows="10" cols="65" class="textarea"	不同版本的特点
Submit	submit	value="添加" class="btn_grey"	提交表单

savebccd.php 文件实现对表单中提交的数据进行处理，首先通过$_POST 获取表单中提交的数据，然后判断指定的文件夹是否存在，最后将数据存储到指定的数据表中。关键代码如下：

例程22　代码位置：光盘\TM\02\bcty365\admin\savebccd.php

```
<?php include_once("../conn/conn.php");                   //连接数据库
$bccdname=$_POST[bccdname];                               //获取 POST 方法提交的值
$owner=$_POST[owner];
$typeid=$_POST[typeid];
$content=$_POST[content];
$samepart=$_POST[samepart];
$addtime=date("Y-m-j H:i:s");                             //获取当前时间
❶    if(is_dir("./bccdimages")==false){                   //判断指定的文件是否存在
❷    mkdir("./bccdimages");                               //如果不存在，则创建一个新的文件夹
}
$link=date("YmjHis");                                     //获取当前时间
//为表单中提交的数据重新命名，以当前时间和随机数作为名称，其中使用$_FILES 获取表单中真实的名称，使用
strstr()函数获取文件的后缀
❸    $path=$link.mt_rand(1000000,9999999).strstr($_FILES["imageaddress"]["name"],".");
$address="./bccdimages/".$path;                           //定义文件上传的路径
move_uploaded_file($_FILES["imageaddress"]["tmp_name"],$address);   //将文件上传到指定的文件中
$imageaddress="./admin/bccdimages/".$path;                //获取上传文件在服务器中的存储路径
//将表单中提交的数据存储到数据库中
$query=mysql_query("insert into tb_bccd(bccdname,owner,typeid,content,samepart,imageaddress,addtime)
values('$bccdname','$owner','$typeid','$content','$samepart','$imageaddress','$addtime')",$conn);
if($query==true){
      echo "<script>alert('编程词典添加成功!');history.back();</script>";
}else{
      echo "<script>alert('编程词典添加失败!');history.back();</script>";
}
?>
```

代码贴士

❶ is_dir()：判断指定的文件是否存在。

❷ mkdir()：创建一个新的文件夹。

❸ mt_rand()：获取随机数字。

strstr()：获取一个指定字符串在另一个字符串中首次出现的位置到后者末尾的子字符串。如果执行成功，则返回剩余字符串（存在相匹配的字符）；如果没有找到相匹配的字符，则返回 false。

2.13.4　编辑编程词典的实现过程

编辑编程词典使用的数据表：tb_bccd、tb_bb、tb_bbqb

在完成对编程词典信息的添加后，接下来就可以对编程词典的版本信息进行编辑，主要添加版本信息、价格、简介、功能和推出的服务。该模块的运行结果如图 2.47 所示。

图 2.47　编辑编程词典模块的运行结果

该功能的实现同样通过两个文件：一个是提交表单的文件 editbccd.php；另一个是处理表单提交数据的文件 sacvbccdbb.php。提交表单文件 editbccd.php 中使用的表单元素如表 2.8 所示。

表 2.8　编辑编程词典页中使用的重要表单元素

名　称	元素类型	重要属性	含　义
form1	form	method="post" action="savebccdbb.php" onSubmit="return chkinput(this)">	编辑编程词典表单
bccdname	text	class="txt_grey" disabled="disabled" value=" <?php $sql4=mysql_query("select bccdname from tb_bccd where id='".$_GET[bccdid]."'",$conn); $info4=mysql_fetch_array($sql4); echo unhtml($info4[bccdname]);　?>	编程词典名称，这里设置了文本的只读属性
bccdid	hidden	value="<?php echo $_GET[bccdid];?>	编程词典的 ID
bbid	select	<?php　$sql3=mysql_query("select * from tb_bb order by createtime desc ",$conn); $info3=mysql_fetch_array($sql3); if($info3==false){ echo　"<option>暂无版本信息</option>"; }else{ do{　?> <option　value="<?php echo $info3[id];?>"><?php echo unhtml($info3[bbname]);?></option> <?php }while($info3=mysql_fetch_array($sql3)); }?>	选择编程词典的版本
Submit	submit	value="添加" class="btn_grey"	提交表单

sacvbccdbb.php 文件对表单提交的数据进行处理，首先获取表单中提交的数据，然后判断指定的版本是否已经被添加，最后将数据存储到指定的数据表中。代码如下：

例程23　代码位置：光盘\TM\02\bcty365\admin\savebccdbb.php

```
<?php
$bccdid=$_POST[bccdid];                                    //获取表单中提交的数据
$bbid=$_POST[bbid];                                        //获取编程词典 ID
$price=$_POST[price];                                      //获取编程词典单价
$content=$_POST[content];                                  //获取编程词典内容
$gn=$_POST[gn];                                            //获取编程词典功能
$fw=$_POST[fw];                                            //获取编程词典服务
include_once("../conn/conn.php");                          //连接数据库文件
//判断提交的编程词典是否已经被添加
$sql=mysql_query("select id from tb_bbqb where bccdid='".$bccdid."'",$conn);
$info=mysql_fetch_array($sql);                             //检索指定编程词典的 ID
if($info!=false){                                          //如果检索值为假，则弹出提示
    echo "<script>alert('该版编程词典已经添加!');history.back();</script>";
    exit; }
/* ****************************************************************** */
$query=mysql_query("insert into tb_bbqb(bccdid,bbid,price,content,gn,fw) values('$bccdid','$bbid', '$price',
'$content','$gn','$fw')",$conn);                           //将表单中提交的数据存储到数据库中
//更新编程词典的价格
$querys=mysql_query("update tb_bccd set bbid='$bbid',price='$price' where id='".$bccdid."'");
if($query==true and $querys==true){                        //如果添加和更新操作为真，则弹出提示
    echo "<script>alert('版本信息添加成功!');history.back();</script>";
}else{                                                     //如果添加和更新操作为假，则弹出提示
    echo "<script>alert('版本信息添加失败!');history.back();</script>";
}
?>
```

2.14　软件升级管理模块设计

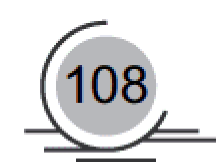
2.14.1　软件升级管理模块概述

软件升级管理模块实现对软件升级包的管理，其具体的功能包括添加升级包、编辑升级包、添加序列号和编辑序列号。软件升级管理模块中的添加升级包和添加序列号是一一对应的，其中根据所属的类别和版本来确定升级包对应的序列号，每一个版本和类别的升级包对应一个序列号。

2.14.2　软件升级管理模块技术分析

在软件升级包管理模块中，应用到一个动态输出下拉列表框中值的技术。下面就来讲解一下该技术是如何实现的，在讲解该技术之前，先来了解下拉列表框的基本结构。

```
<select name="select"><!--name 指定该下拉列表框的名称-->
    <!--selected 设置下拉列表框的默认值，默认值为 PHP-->
    <option selected="selected">PHP</option>
```



```
	<!--value 指定的 mysql 是下拉列表框传递的值，MYSQL 为显示的内容-->
	<option value="mysql">MYSQL</option>
</select>
```

所谓动态输出下拉列表框中的值就是从数据库中读取数据，将获取到的数据输出到下拉列表框中，而不是直接在下拉列表框中设置某个固定的值。这里以软件升级管理模块 addsjb.php 文件中的所属类别下拉列表框为例进行讲解，其中设置下拉列表框的名称为 typeid，默认值为“请选择”，value 的值是从数据库中获取的 ID 值，显示的内容为从数据库中获取的类型名称。动态输出下拉列表框中的值使用的关键代码如下：

例程24　代码位置：光盘\TM\02\bcty365\ admin\wzdh.php

```
<!-- ----------------------------------设置下拉列表框的名称为 typeid--------------------------------- ->
<select name="typeid" class="txt_grey">
	<option value="" selected="selected">请选择</option>
	<!-- ------------------------------------设置下拉列表框的名称为 typeid------------------------ -->
	<?php
		include_once("../conn/conn.php");		//连接数据库
		//从数据库中读取编程词典类型的数据
		$sql=mysql_query("select * from tb_type order by createtime desc",$conn);
		$info=mysql_fetch_array($sql);
		if($info==false){
			echo "<option >暂无类别</option>";
		}else{
			do{						//应用 do…while 循环语句输出类型的 ID 和类型的名称
				echo "<option value=".$info[id].">".$info[typename]."</option>";
			}
			while($info=mysql_fetch_array($sql));	//do…while 循环语句结束
		}
	?>
</select>
```

下拉列表框不但可以动态输出数据库中某个字段的数据，而且可以输出数组中的数据。下面就实现一个在下拉列表框中动态输出数组中数据的功能，首先创建一个下拉列表框，然后设置下拉列表框的值，从数组中读取数据，应用 for 循环语句进行输出。代码如下：

```
<!-- ------------------------------设置下拉列表框的名称为 select---------------------------------- -->
<select name="select" size="1">
<?php
	$string="ASP,PHP,JSP,.NET,DEL,VB,VC";		//定义一个字符串
	$srtings=split(",",$string);			//对字符串进行分割
	$count=count($srtings);				//获取数组中元素的数量
	for($i=0;$i<$count;$i++){			//根据数组中元素的数量进行循环输出
		$result=$srtings[$i];			//定义变量，获取数组中指定的元素
	echo "<option>$result</option>";}		//将数组中的元素输出到下拉列表框中
?>
</select>
```

动态输出数据库中的数据到下拉列表框的运行结果如图 2.48 所示。

图 2.48　动态输出数据库中的数据到下拉列表框

2.14.3　软件升级包上传的实现过程

软件升级包上传使用的数据表： tb_bb、tb_type、tb_sjxz

软件升级包上传在添加升级包模块中实现，通过一个文件域文本框将升级包提交到服务器中指定的文件下，并且将该文件在服务器中的路径存储到数据库中，便于在前台实现对软件升级包的下载。其运行结果如图 2.49 所示。

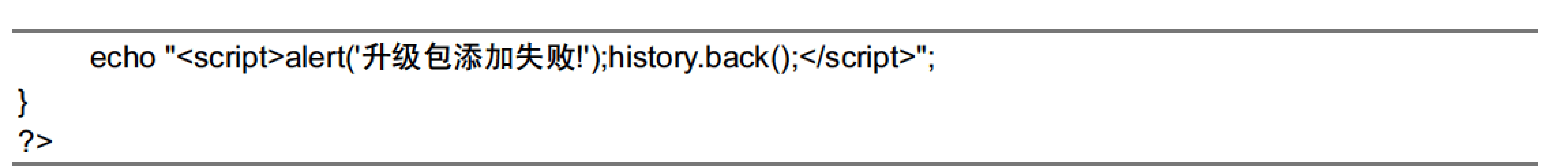


图 2.49　软件升级包上传的运行结果

在本模块中通过 addsjb.php 文件来提交升级包的信息，通过 savesj.php 文件来对表单提交的数据进行处理。其中在将升级包上传到服务器的指定文件夹的过程中，主要应用的是 move_uploaded_file()函数。在 savesj.php 文件中，首先获取表单提交的数据，然后判断服务器中是否存在指定的文件，最后应用 move_uploaded_file()函数将升级包上传到指定的文件夹下，并且将数据存储到指定的数据表中。程序代码如下：

倒程25　代码位置：光盘\TM\02\bcty365\admin\savesj.php

```
<?php
$name=$_POST[name];                              //获取表单提交的数据
$typeid=$_POST[typeid];                          //获取表单提交的数据
$content=$_POST[content];                        //获取表单提交的数据
$addtime=date("Y-m-j H:i:s");                    //定义时间变量
$bbid=$_POST[bbid];                              //获取表单提交的数据
if(is_dir("./sjxz")==false){                     //判断指定的文件夹是否存在
    mkdir("./sjxz");                             //如果指定的文件夹不存在，则创建一个指定的文件夹
}
$link=date("YmjHis");                            //获取一个时间
$path=$link.mt_rand(1000000,9999999).strstr($_FILES["address"]["name"],".");//重新设置升级包名称
$address="./sjxz/".$path;                        //设置升级包在服务器中存储的指定路径
move_uploaded_file($_FILES["address"]["tmp_name"],$address);    //将升级包上传到指定的路径下
$address="./admin/sjxz/".$path;                  //获取升级包在服务器中的存储路径
include_once("../conn/conn.php");                //连接数据库文件
//将上传的数据存储到数据库中，这里将升级包在服务器中的路径存储到数据库中
$query=mysql_query("insert into tb_sjxz(name,typeid,content,addtime,address,bbid)
values('$name','$typeid','$content','$addtime','$address','$bbid')",$conn);
if($query){                                      //如果添加操作成功，则弹出提示
    echo "<script>alert('升级包添加成功!');history.back();</script>";
}else{                                                          //如果添加操作失败，则弹出提示
```



```
    echo "<script>alert('升级包添加失败!');history.back();</script>";
}
?>
```

2.14.4　软件升级包删除的实现过程

软件升级包删除使用的数据表：tb_bb、tb_type、tb_sjxz

软件升级包删除的实现主要根据当前数据中提供的 ID，执行 delete 删除语句，将数据表中相同 ID 的数据删除。其运行结果如图 2.50 所示。



图 2.50　软件升级包删除的运行结果

该功能主要通过 editsjb.php 文件和 deletesjb.php 文件实现。通过 editsjb.php 文件输出数据库中存储的有关升级包的信息，以分页的形式显示，在每条记录的最后设置一个删除链接，通过脚本来调用 deletesjb.php 文件，根据变量中的 ID 值执行删除升级包的操作。关键代码如下：

例程26　代码位置：光盘\TM\02\bcty365\admin\deletesjb.php

```
<?php
$id=$_GET[id];                                                          //获取变量传递的 ID
include_once("../conn/conn.php");                                       //连接数据库
//执行删除操作，将数据表中对应的 ID 数据删除
if(mysql_query("delete from tb_sjxz where id='".$id."'",$conn)){
    echo "<script>alert('该升级包删除成功!');history.back();</script>";    //如果删除操作成功，则弹出提示
}else{                                                                  //如果删除操作失败，则弹出提示
    echo "<script>alert('该升级包删除失败!');history.back();</script>";
}
?>
```

2.15　在 Linux 系统下发布网站

在 Linux 系统下发布基于 PHP 的网站，首先需要配置 PHP 的运行环境，其次需要对网卡参数进行设定。这里将以发布“BCTY365 网上社区”网站为例讲解在 Linux 系统下如何实现网站的发布。假设已经申请到表 2.9 所示的网络参数。

表 2.9　申请到的网络参数

参　数	值
IP	192.168.1.*
Netmask	255.255.255.*
Network	192.168.1.0
Broadcast	192.168.1.*
Gateway	192.168.1.*
主机名	Tsoft
DNS	168.95.1*

在 Linux 系统下网站发布的操作步骤如下：

（1）配置 PHP 的运行环境，在 2.4 节中已经做了详细介绍，这里不再赘述。

（2）将“BCTY365 网上社区”网站的所有文件复制到 Apache 主目录下。

（3）设置主机名称。在终端窗口中输入如下命令编辑/etc/sysconfig/network 文件：

```
vi /etc/sysconfig/network
```

将该文件中的参数 NETWORKING 设置为 yes，表示启动网络，将参数 HOSTNAME 设置为 Tsoft，表示设置主机名为 Tsoft。

（4）设置网卡参数。在终端窗口中输入如下命令编辑文件/etc/sysconfig/network-scripts/ifcfg-eth0：

```
vi   /etc/sysconfig/network-scripts/ifcfg-eth0
```

该文件的相关参数设置如表 2.10 所示。

表 2.10　设置网卡的相关参数

参　数	说　明
DEVICE=eth0	设置网卡名称，要与 ifcfg-eth0 对应
ONBOOT=yes	指定在开机时启动网卡
BOOTPROTO=static	设定启动时获取 IP 的方式
IPADDR=192.168.1.*	设定服务器 IP 地址
NETMASK=255.255.255.*	设定子网掩码
BROADCAST=192.168.1.*	设定同网段的广播地址
GATEWAY=192.168.1.*	设定网卡的网关

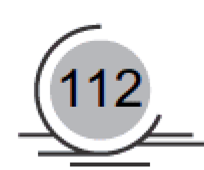
（5）设置 DNS 主机的 IP。在终端编辑/etc/resolv.conf 文件：

```
vi /etc/resolv.conf
```

设置参数 nameserver 的值为 168.95.1.*。

（6）重新启动网络设置。在终端窗口中输入如下命令：

```
/etc/rc.d/inin.d/network restart
ifdow eth0
ifup eth0
```





（7）打开浏览器，在地址栏中输入服务器 IP 地址或域名，打开如图 2.51 所示页面，则说明在 Linux 系统下“BCTY365 网上社区”网站发布成功。



图 2.51　在 Linux 系统下“BCTY365 网上社区”网站运行结果

2.16　开发技巧与难点分析

2.16.1　管理员权限的设置

为了更好地管理和维护网站的论坛，针对论坛设置了一个管理员，该管理员不在后台进行操作，而是在前台为管理员设置特殊的权限，也可以称之为版主。其实现的原理是：首先在数据库中设置不同的值代表不同的权限，0 代表普通会员，1 代表管理员；然后在论坛的页面中进行判断，当用户的类型为 1 时，不但具有普通会员的权限，而且具有删除发布帖子、回复帖子和顶帖的权限；如果用户的类型不是 1，则不具有上述的权限，只能是发布和回复帖子。管理员和普通会员登录的页面效果是不同的，如图 2.52 和图 2.53 所示。



图 2.52　管理员登录的操作页面



图 2.53　普通会员登录的操作页面

在页面中执行的判断语句判断登录用户的类型，然后根据类型判断用户的权限。程序关键代码如下：

```
<?php
    if($_SESSION["unc"]!=""){                                    //判断 session 变量的值是否为空
        //根据 session 变量的值获取该用户的类型
        $sqlu=mysql_query("select usertype from tb_user where usernc='".$_SESSION["unc"]."'",$conn);
        $infou=mysql_fetch_array($sqlu);
```

```
            if($infou["usertype"]==1){                          //判断当该用户的类型等于 1 时执行下面的内容
?>
<!--调用 JavaScript 脚本，执行删除发布帖子的操作-->
<img src="images/lt_15(10).jpg" onclick="javascript:if(window.confirm('您确定删除该帖么？
')==true){window.location.href='bbs_delete.php?id=<?php echo $infob["id"]?>';}" style="cursor:hand"/>
<?php
            }
            //如果用户的类型不是 1，则不执行上述的内容
        }
?>
```

2.16.2 帖子置顶的设置

所谓帖子置顶就是将某个指定的帖子在对应的版块中最前面的位置显示，该权限只有管理员才拥有，普通会员不具备该权限。其实现的原理如下：

首先，在数据库存储发布帖子信息的数据表中设置一个字段 top，指定该字段为数字类型，其默认值为 0。

然后，在网页中判断登录用户的权限，如果是管理员，则具有帖子置顶的权限，否则将弹出提示对话框"对不起，您不具备该操作权限！"。

最后，如果是管理员，则执行 settop.php 文件，根据对应帖子的 ID 查找到发布帖子信息表中对应的数据，更新该条数据中 top 字段的值为 1。

判断登录用户权限使用的代码如下：

```
<?php
    if($_SESSION["unc"]==""){                   //判断 session 变量的值是否为空，如果为空则执行下面的脚本
        echo "javascript:alert('请先登录本站，然后进行此操作!');window.location.href='index.php';";
    }else{
        //如果不为空则执行下面的内容，判断登录用户的权限
        $sqlu=mysql_query("select usertype from tb_user where usernc='".$_SESSION["unc"]."'",$conn);
        $infou=mysql_fetch_array($sqlu);
        if($infou["usertype"]==1){              //如果登录用户的类型是 1，则说明是管理员，执行下面的内容
            echo "javascript:window.location.href='settop.php?id=".$infob["id"]."'";
        }else{
            //如果登录用户不是管理员，则执行下面的内容
            echo "javascript:alert('对不起，您不具备该操作权限!');";
        }
    }
?>
```

实现帖子置顶是通过 settop.php 文件来完成的，在该文件中，根据变量提交的值获取到发布帖子信息表中对应的数据，更新该条数据中字段 top 的值，并且对该字段的值进行判断。如果字段 top 的值为 1，则说明该帖已经置顶，此时将字段的值更新为 0，即取消置顶；如果字段 top 的值为 0，则说明该帖没有进行置顶，此时将字段的值更新为 1，即置顶该帖。settop.php 文件的程序代码如下：

```
<?php   include_once("conn/conn.php");                               //连接数据库文件
//根据获取的 ID 值，从数据表中读取到对应的数据
```



```
$sql=mysql_query("select top from tb_bbs where id='".$_GET["id"]."'",$conn);
$info=mysql_fetch_array($sql);
if($info[top]==1){          //判断对应数据记录中字段 top 的值，如果字段 top 的值为 1，则执行下面的内容

    mysql_query("update tb_bbs set top=0 where id='".$_GET["id"]."'",$conn);//更新字段 top 的值为 0
}elseif($info[top]==0){      //如果对应数据记录中字段 top 的值为 0，则执行下面的内容
    mysql_query("update tb_bbs set top=1 where id='".$_GET["id"]."'",$conn);//更新字段 top 的值为 1
}
echo "<script>alert('置顶设置成功!');history.back();</script>";
?>
```

2.17　在线支付技术专题

所谓在线支付就是客户端（金融机构需客户端安装由金融机构签发的数字证书，信用卡免安装）将支付信息加密后通过互联网传送到支付网关（支付网关是解决网络上安全支付问题的交易平台，位于互联网和传统的金融机构内部网之间，其主要作用是将互联网和金融网络安全地连接起来，将不安全的网上交易信息传给安全的金融网络，起到隔离和保护金融网络的作用），同时金融机构网上支付系统反馈有关支付信息，客户确认无误后进行支付确定，支付网关负责商户网上交易资金的清算，并根据商户提供的开户行、账号等结账信息将网上消费款项汇总划入商户账户。

BCTY365 网上社区的在线支付是与中国工商银行合作来共同完成的。BCTY365 网上社区的在线支付操作步骤如下：

（1）登录网上社区，如图 2.54 所示。

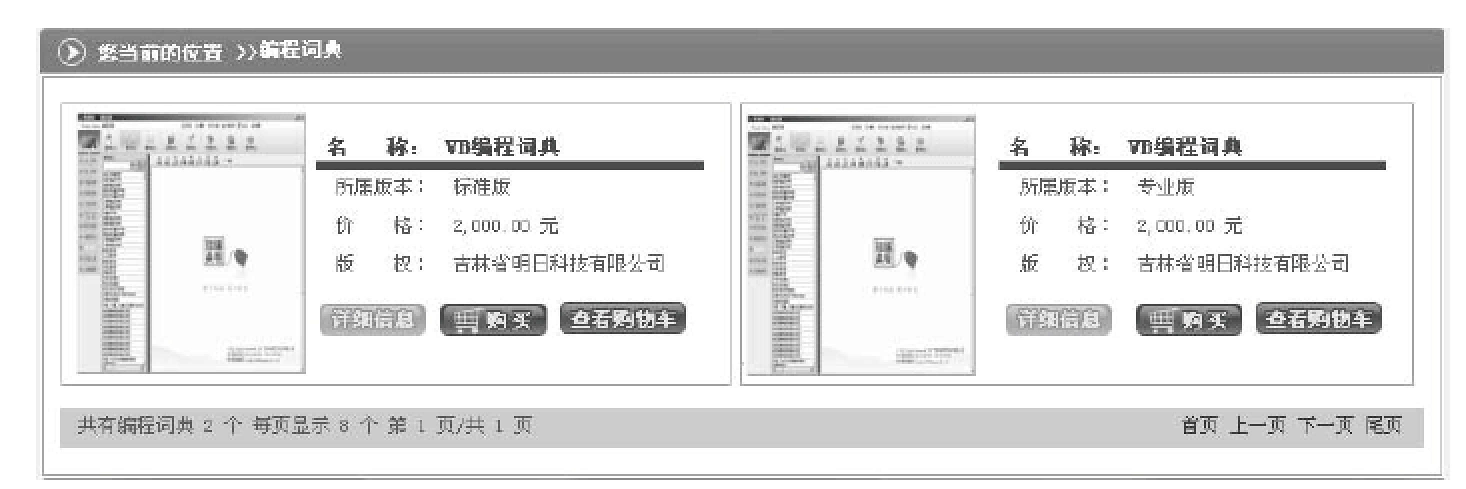


图 2.54　在线订购的操作页面

（2）购买商品。在本页中，不但可以购买商品，还可以查看商品的详细信息和购物车中的商品信息，如图 2.55 所示。

（3）进入购物车操作页面。在该页面中，可以修改购物数量、删除指定商品、清空购物车、继续购物和统计购买商品的金额，也可以单击“结算”按钮进入到商品结算页面，如图 2.56 所示。



图 2.55　购买商品操作页面



图 2.56　购物车操作页面

（4）进入到购物结算页面，填写收货人的详细信息，确认后提交该数据，如图 2.57 所示。

（5）订单确认。订单确认以后，就可以提交订单，准备进行网上支付，如图 2.58 所示。



图 2.57　填写收货人的详细信息



图 2.58　订单确认

（6）进行网上支付。在这里可以选择工行网上支付，也可以选择取消该订单，如图 2.59 所示。

订单号：07121311451945　　需支付金额：2240.1 元
＊ 只有在网上支付成功后，我公司才会为您邮递。
＊ 我们会在48小时内保留您的订单，请及时支付。
取消订购　工行网上支付

图 2.59　执行网上支付

接下来的操作在工行 B2C 支付页面上进行。首先网上社区按照工商银行 B2C 订单数据规范形成提交数据，并使用工商银行提供的 API 和商户证书对订单数据签名，形成 form 表单返回客户浏览器，表单 action 地址指向工商银行接收商户 B2C 订单信息的 servlet；然后在客户确认使用工行网上支付后，提交此表单到工商银行；最后工行网银系统接收此笔 B2C 订单，对订单信息和商户信息进行检查，通过检查则显示工行 B2C 支付页面。

客户通过工行 B2C 支付页面实现网上支付，商户查询网上银行的账户，如果货款已经到账，则根据客户指定的方式将货物送达客户手中。

上述内容就是网上社区系统的在线支付流程，涉及工商银行的操作内容这里不做讲解。这里主要讲解一下如何将订单信息提交到工商银行。该项操作主要通过 shopping_tjdd.php 文件来实现，首先从数据库中读取订单信息，然后将订单信息进行输出，最后创建“取消订购”和“工行网上支付”两个超链接，通过 JavaScript 脚本来调用不同的执行文件。关键代码如下：

```
<?php    include_once("conn/conn.php");     include_once("top.php");  //连接数据库和网站的头文件 ?>
<!--省略了部分代码-->
<?php
❶    $ddnumber=base64_decode($_GET["ddno"]);                    //对获取的订单编号进行 base64 解码
//获取该订单的金额信息
$sql=mysql_query("select * from tb_dd where ddnumber='".$ddnumber."'",$conn);
$info=mysql_fetch_array($sql);
$amount=$info["totalprice"];
❷    $amount=str_replace(",","",number_format($amount,2));      //修改数字的输出格式
$amount=str_replace(".","",number_format($amount,2));          //修改数字的输出格式
?>
…                                                              //省略部分 HTML 代码
<table width="630" border="0" align="center" cellpadding="0" cellspacing="0">
     <tr><td width="159"> 
<?php
     $sql=mysql_query("select totalprice from tb_dd where ddnumber='".base64_decode($_GET ["ddno"])."'",
$conn);
     $info=mysql_fetch_array($sql);
```



```
            echo "<font color=red><strong>".$info["totalprice"]." 元</strong></font>";
?>          </td>
        </tr>
</table>
…                                                                       //省略部分 HTML 代码
<script language="javascript">
//打印订单
❸      function openprintwindow(x,y,z){
        window.open("printwindow.php?ddno="+x+"&pv="+z,"newframe","top=200,left=200,width=635,hei
ght="+(230+20*y)+",menubar=no,location=no,toolbar=no,scrollbars=no,status=no");
        }
</script>
…                                                                       //省略部分 HTML 代码
<table width="630" height="25" border="0" align="center" cellpadding="0" cellspacing="0">
        <tr>
           <!-- ------------------------------------------取消该订单---------------------------------------- -->
            <td width="75"><img src="images/bg_14(14).jpg" width="69" height="20" style="cursor:hand" onclick=
"javascript:if(window.confirm('如果取消该订单，则该订单将被删除，您需要重新购买！')==true){window. location.
href='deletedd.php?ddno=<?php echo $_GET["ddno"];?>';}"/></td>
           <!-- -----------------------------------------执行网上支付--------------------------------------- -->
            <td width="125"><img src="images/bg_14(15).jpg" width="119" height="20" onclick="javascript:window.
location.href='ddform.php?orderid=<?php echo base64_decode($_GET["ddno"]);?>&amount=<?php echo $amount;?
>&orderDate=<?php echo date("Ymdhis");?>';" style="cursor:hand"/></td>
        </tr>
</table>
<!-- -------------------------------------------------省略了部分代码------------------------------------------ -->
<?php     include_once("bottom.php");                           //包含网站的尾文件          ?>
```

代码贴士

❶ base64_decode()：PHP 实现对 base64 编码的字符进行解码。PHP 实现字符串的 base64 编码通过 base64_encode()函数。

❷ str_replace()：实现字符串的替换。该函数的语法如下：

```
mixed str_replace ( mixed search, mixed replace, mixed subject , int &count )
```

str_replace()将所有在参数 subject 中出现的 search 以参数 replace 替换，参数&count 表示替换字符串执行的次数。

❸ openprintwindow ()：JavaScript 脚本中自定义的函数，用于执行订单的打印操作。

有关在线支付流程中的其他操作实现方式已经在 2.10 节中进行了详细的讲解，这里不再赘述。其具体的代码可参考本书的光盘 TM\02\bcty365\。

2.18　本 章 总 结

本章从项目开发的实际角度出发，以某科技公司的实际需求为背景，详细地讲解“BCTY365 网上社区”系统的开发过程，其中以系统的整体开发流程为主线，重点介绍技术支持、在线订购、社区论坛和编程词典管理等几个大模块的实现方法，并且对管理员权限设置、帖子置顶设置和在线支付技术进行了难点分析和专题讲解，而且在本章中还讲解了在 Linux 系统下如何搭建 PHP 的开发环境以及如何发布网站。

第 3 章

办公自动化管理系统

（Apache+PHP+phpMyAdmin+MySQL 5.0 实现）

办公自动化（Office Automation，OA）系统指实现办公室内事务性业务的自动化。

办公自动化没有明确的定义，最普遍的说法是凡是在传统的办公室中采用各种新技术、新机器、新设备从事办公业务，都属于办公自动化的领域。

办公自动化系统与办公自动化在概念上存在一定的差别。办公自动化通常指办公室中配备具有自动化功能的设备，这些设备能使某些办公活动自动化或实现某个单位业务的自动化处理；而办公自动化系统则是在办公自动化功能的基础上发展起来，以办公自动化技术为主体，同人、组织、制度、环境等相结合的完整的系统。通过阅读本章，可以学习到：

- 办公自动化管理系统的开发流程
- 进一步学习如何做项目需求分析与系统设计
- 页面布局中的框架布局
- 使用递归函数做多级下拉菜单
- 系统日志的实现



3.1 开发背景

随着中国加入 WTO 及全球经济一体化进程的加快，世界经济已由工业化经济逐步进入网络信息化时代。在信息时代来临之际，各企业都紧跟时代的脚步，转变着企业的经营模式、管理模式，从传统的人工管理体制向信息自动化管理体制过渡。网络自动化办公系统就是在这样的大背景下应运而生的。

信息时代的到来让人们尝到了“信息爆炸”的滋味，信息的大量涌入让企业在信息处理方面应接不暇，传统的办公模式对信息的处理方法早已不能满足企业对信息快速、准确处理的要求。一个企业对信息数据的掌握程度、处理能力，体现了企业对市场的敏感程度，数据的真实性、准确性直接决定着企业的发展方向。从传统的办公模式向自动化办公管理模式转变，提高企业的信息处理能力，以增强企业的市场竞争力，成了企业发展过程中的首要问题。

3.2 需求分析

根据与客户的多次交谈和了解，本系统所面向的客户对象情况如下：

☑ 所属 IT 行业，目前主要以开发门户网、企业网等中小型网站为主。
☑ 公司经过多年经营，已经相对稳定，并有了自己的网站、企标、规章制度和基本架构。
☑ 公司规模 50～100 人，人手一机，主要包括部门经理、普通职员、试用人员和实习人员。
☑ 职员所用计算机属于局域网内网，不允许访问外网。
☑ 公司暂时分为技术部、人事部和质量部，但不排除后期增加其他部门的可能。
☑ 公司实行人性化管理，允许员工自由发表意见和想法，还有丰富多彩的活动，如比赛、旅游等。
☑ 为了提高工作效率，所有职员定期写工作计划，包括周计划、月计划，部门经理还有年计划和任务计划。
☑ 根据计划的完成程度和完成质量，不定期地选取优秀员工。

3.3 系统设计

3.3.1 系统目标

本系统是针对中小型企业内部自动化办公管理的要求进行设计的，主要实现如下目标：

☑ 键盘操作，快速响应。
☑ 实现文件类信息的强大的管理能力。
☑ 实现对员工基础信息（人事消息）的管理功能等。
☑ 实现个人办公的信息自动化管理功能。
☑ 发布会议信息，并对会议信息进行管理。

☑ 对系统用户进行管理。

☑ 为了加强数据保密性，为每个用户组设置权限级别。

☑ 系统最大限度地实现了易安装性、易维护性和易操作性。

☑ 系统运行稳定、安全可靠。

3.3.2 系统功能结构

根据系统分析，下面给出系统的前、后台功能结构图。

办公自动化前台功能结构图如图 3.1 所示。

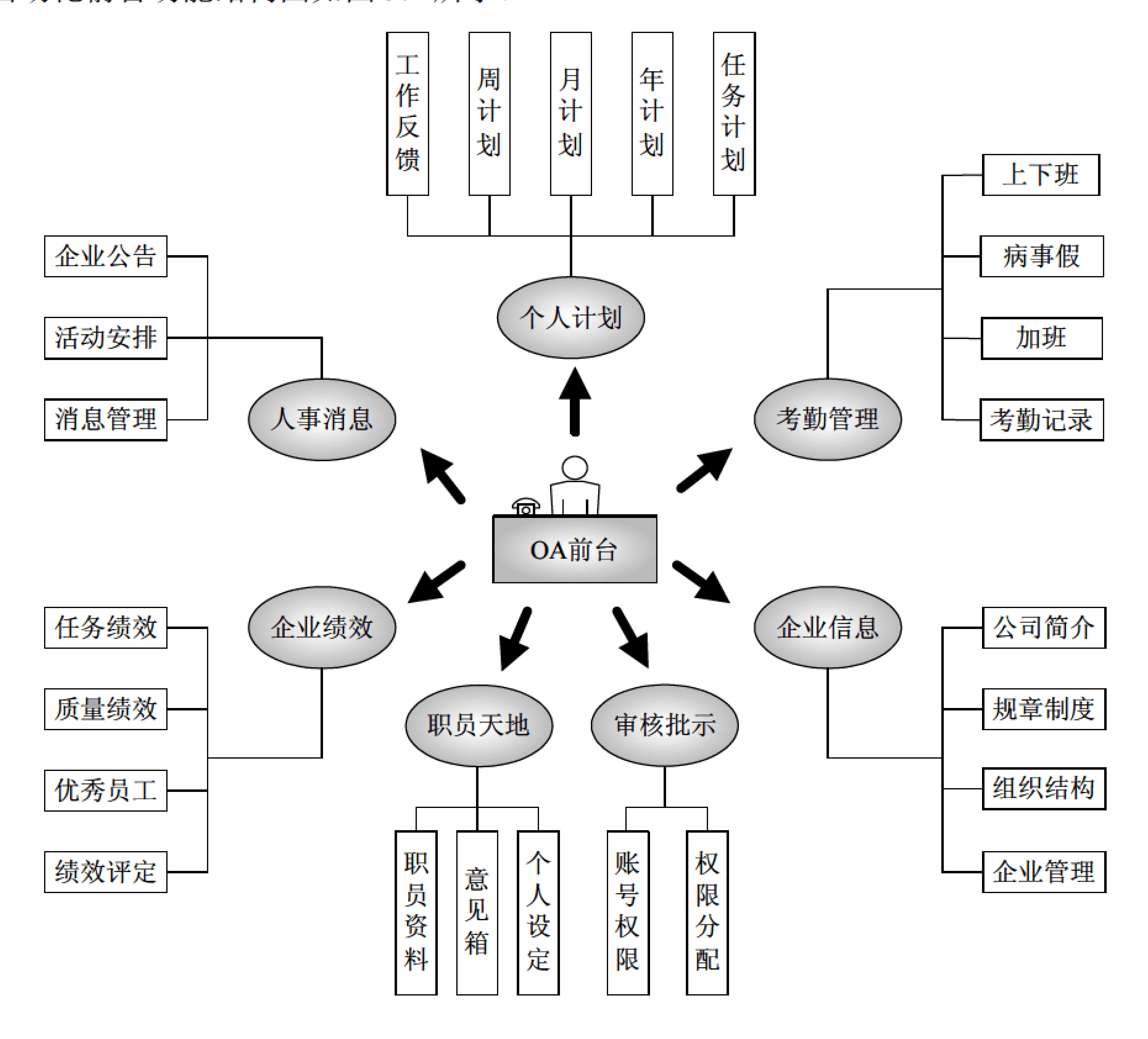


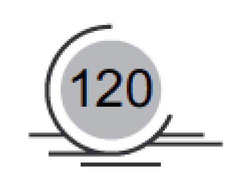
图 3.1 办公自动化前台管理系统

办公自动化后台功能结构图如图 3.2 所示。

3.3.3 系统功能预览

办公自动化管理系统由多个功能模块组成，下面仅列出几个典型功能的页面，其他页面参见光盘中的源程序。

前台登录界面如图 3.3 所示，该页面用于实现对用户登录的用户名和密码进行验证。企业信息页面如图 3.4 所示，该页面用于显示企业文化和各规章制度。





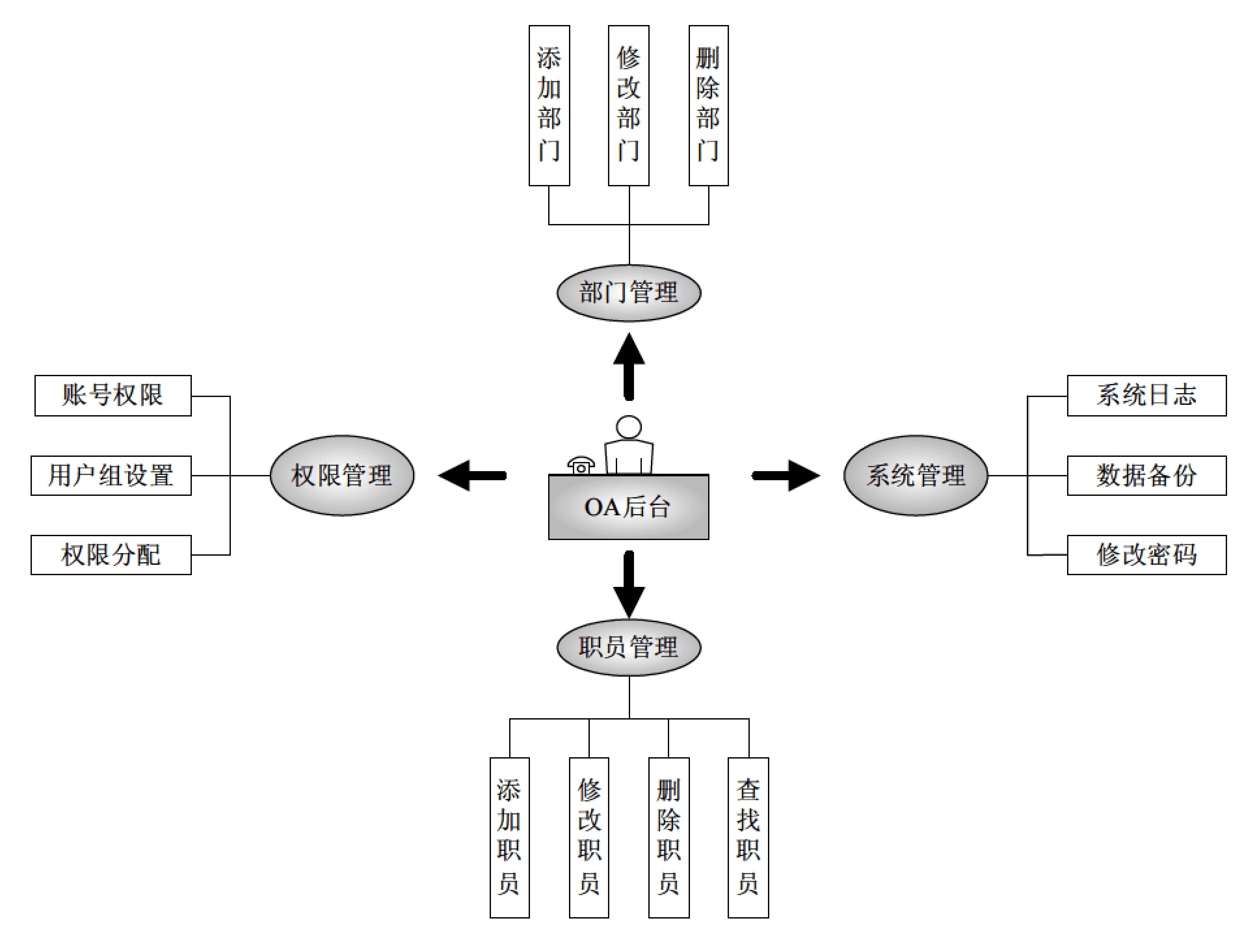


图 3.2　办公自动化后台管理系统





图 3.3　前台登录（光盘\TM\03\oa\index.php）

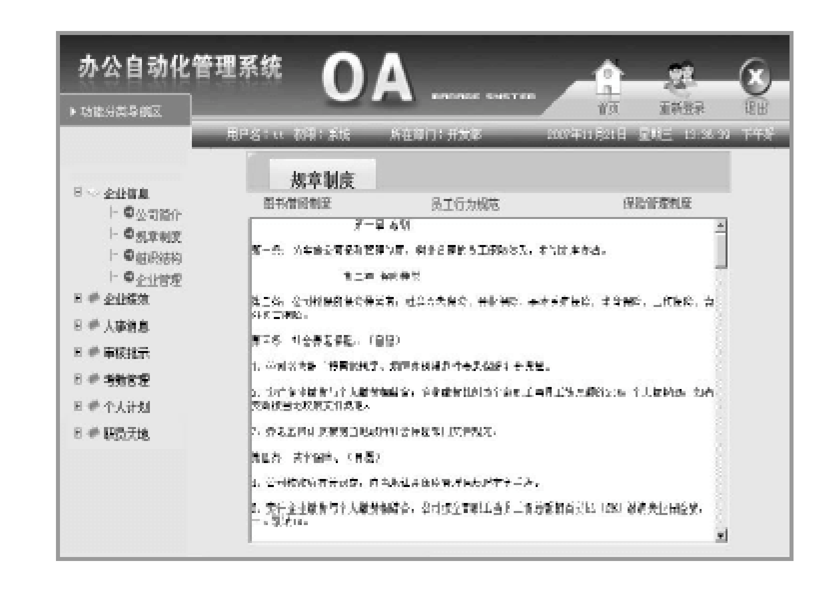


图 3.4　企业信息（光盘\TM\03\oa\qyxx\r_system.php）

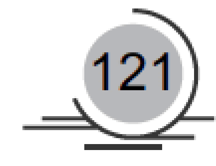
个人计划页面如图 3.5 所示，该页面主要实现用户的工作反馈和周、月计划等功能。后台登录界面如图 3.6 所示，该页面用于管理员后台登录。



图 3.5　个人计划（光盘\TM\03\oa\grjh\person_plan.php）



图 3.6　后台登录（光盘\TM\03\oa\admin\index.php）



职员管理页面如图 3.7 所示，该页面主要用于实现对职员的查询与修改等功能。权限分配页面如图 3.8 所示，该页面用于对功能使用权的分配。





图 3.7　职员管理（光盘\TM\03\oa\admin\zygl\show_staf.php）　图 3.8　权限分配（光盘\TM\03\oa\admin\qxgl\pur_assign.php）

3.3.4　系统流程图

用户在登录一个系统后会进行一系列的操作，把这些操作的过程和结果以图形的形式表现出来，这就是系统流程图。一个好的流程图，不仅可以让开发者迅速地理清思路、及时解决出现的问题，也可以让使用者很快明白该系统的操作方式与方法。办公自动化管理系统的工作流程图如图 3.9 所示。

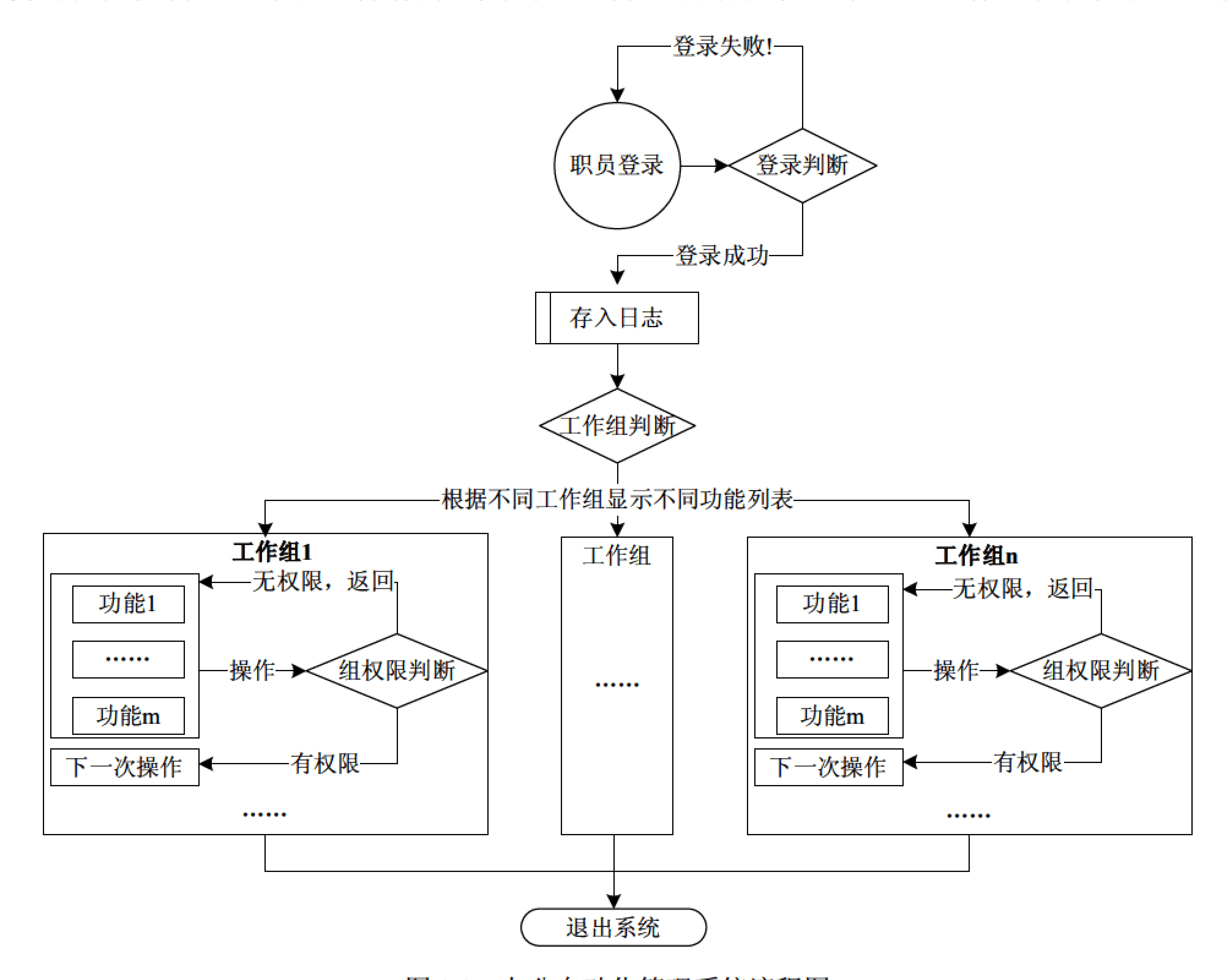


图 3.9　办公自动化管理系统流程图



3.3.5　开发环境

在开发办公自动化管理系统平台时，该项目使用的软件开发环境如下。

1．服务器端

- ☑ 操作系统：Windows Server 2003。
- ☑ 服务器：Apache 2.2.8。
- ☑ PHP 软件：PHP 5.2.6。
- ☑ 数据库：MySQL 5.0.51。
- ☑ MySQL 图形化管理软件：phpMyAdmin-2.10.3。
- ☑ 开发工具：Dreamweaver 8。
- ☑ 浏览器：IE 6.0。
- ☑ 分辨率：最佳效果为 1024×768 像素。

2．客户端

- ☑ 浏览器：推荐使用 IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

3.3.6　文件夹组织结构

系统功能结构和系统流程设计完成后，接下来设计网站的文件夹结构，合理的文件夹结构不仅易于快速地开发，对后期系统的调试、维护和管理也能起到事半功倍的效果。本系统的文件夹组织结构如图 3.10 所示。



图 3.10　办公自动化管理系统文件夹组织结构

3.4　数据库设计

基于目前的系统开发，没有数据库的支持根本是无法想象的，办公自动化管理系统更是使用了大量的数据表来存储数据。本节就对办公自动化管理系统所需要使用的数据库进行说明与设计。

3.4.1　数据库分析

由于本系统采用的是 PHP 语言，数据库理所当然地使用 MySQL，不仅是因为开发成本低，更重要的是两者之间的默契程度和稳定程度要远远高于其他的数据库组合，并且 MySQL 数据库对于一个企业的内部办公自动化管理系统完全够用。

3.4.2　数据库概念设计

根据以上的需求分析及系统分析，规划出办公自动化管理系统主要的几个实体关系 E-R 图。

1. 用户信息实体

用户信息实体包括职员的账号、密码、姓名、电话等表示个人身份的数据资料，如图 3.11 所示。

2. 考勤登记实体

考勤登记实体包括登记日期、登记时间、登记类别（上班或下班）、登记状态（迟到或早退）和登记人、备注等信息，如图 3.12 所示。

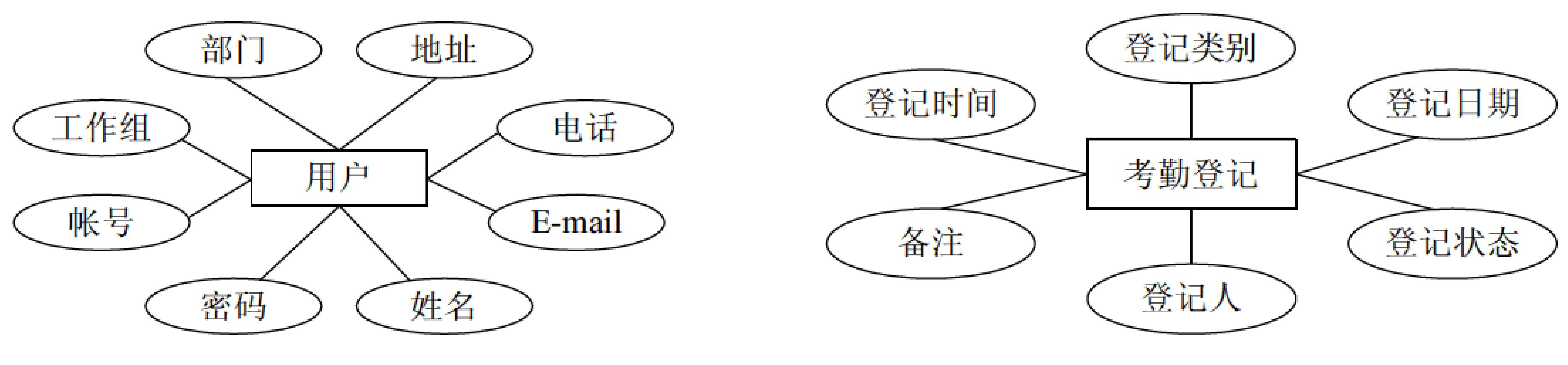


图 3.11　用户信息实体 E-R 图

图 3.12　考勤登记实体 E-R 图

3.4.3　数据库物理结构设计

在本系统中创建了一个数据库 db_office，一共包含 13 个数据表，如图 3.13 所示。

限于篇幅，这里只给出主要的表结构和表说明，其他数据表结构请参见本书附带的光盘。

服务器: localhost ▸ 数据库: db_office

表	类型	整理	说明
tb_company	MyISAM	gb2312_chinese_ci	企业信息列表
tb_controller	MyISAM	gb2312_chinese_ci	管理员列表
tb_depart	MyISAM	gb2312_chinese_ci	部门管理列表
tb_group	MyISAM	gb2312_chinese_ci	用户组管理列表
tb_iss	MyISAM	gb2312_chinese_ci	审核列表
tb_list	MyISAM	gb2312_chinese_ci	功能列表
tb_lyb	MyISAM	gb2312_chinese_ci	意见箱列表
tb_person	MyISAM	gb2312_chinese_ci	人事列表
tb_plan	MyISAM	gb2312_chinese_ci	计划列表
tb_register	MyISAM	gb2312_chinese_ci	登记列表
tb_setup	MyISAM	gb2312_chinese_ci	时间列表
tb_superson	MyISAM	gb2312_chinese_ci	优秀员工列表
tb_users	MyISAM	gb2312_chinese_ci	用户列表

图 3.13　db_office 数据库中的数据表列表

1. tb_users（用户列表）

用户列表主要用于存储职员的姓名、性别等私人信息及部门、工作组等与公司相关的公共信息。该数据表结构如图 3.14 所示。

2. tb_person（人事列表）

人事列表用于存储人事部门发布的信息，如信息标题、信息内容、信息时间等。该数据表结构如图 3.15 所示。

服务器: localhost ▸ 数据库: db_office ▸ 表 : tb_users

字段	类型	整理	属性	Null	默认	额外	说明
id	int(4)			否		auto_increment	自动编号
u_user	varchar(50)	gb2312_chinese_ci		否			用户账号
u_pwd	varchar(20)	gb2312_chinese_ci		否			用户密码
u_name	varchar(20)	gb2312_chinese_ci		是	NULL		用户姓名
u_sex	char(1)	gb2312_chinese_ci		是	NULL		用户性别
u_birth	date			是	NULL		用户生日
u_address	varchar(50)	gb2312_chinese_ci		是	NULL		用户住址
u_tel	varchar(20)	gb2312_chinese_ci		是	NULL		用户电话
u_email	varchar(50)	gb2312_chinese_ci		是	NULL		用户Email
u_depart	varchar(20)	gb2312_chinese_ci		否			所属部门
is_on	int(1)			否			激活账号

图 3.14　用户列表

服务器: localhost ▸ 数据库: db_office ▸ 表 : tb_person

字段	类型	整理	属性	Null	默认	额外	说明
id	int(4)			否		auto_increment	自动编号
p_title	varchar(50)	gb2312_chinese_ci		否			公告标题
p_content	mediumtext	gb2312_chinese_ci		否			公告内容
p_time	date			否			发布时间
u_id	int(4)			否			消息类别

图 3.15　人事列表





3.5　公共模块设计

本系统的公共模块包括 conn 数据库链接文件、css 样式文件、JavaScript 脚本文件以及 inc 下的自定义函数文件和包含文件等几类文件，其中数据库链接文件和 css 样式文件在前面的章节中有过系统的介绍，这里主要讲解本系统所涉及的 JavaScript 脚本文件和部分自定义函数文件。

3.5.1　JavaScript 脚本

在办公自动化管理系统中，JavaScript 脚本一般用于表单元素验证，如判断 text 文本框输入是否为空，输入格式是否符合标准等。在网页中使用 JavaScript 脚本的方式主要有以下 3 种：

（1）在网页中使用<script></script>标签对

<script></script>标签对可以放在网页的任意位置，一般放在<head></head>或<body></body>之间。代码如下：

```
<head>
<script>document.write("办公自动化管理系统开发");</script>
</head>
```

（2）在单独文件中使用

如果 JavaScript 脚本比较多，而且位置分散不易管理，可以统一放到一个扩展名为 js 的文件中，使该文件成为 JavaScript 脚本文件。在脚本文件中，不需要使用<script></script>标签对，直接写脚本代码即可。当有页面需要使用到里面的 JavaScript 脚本时，可以这样引用，代码如下：

```
<script src="ad_js.js" jangauge="javascript"></script>
```

（3）在表单元素或标签中使用

这是最直接的使用方式，如果是少量的脚本则可以这样使用。例如，在超链接标签<a>中想要使用 JavaScript 脚本，代码格式如下：

```
<a href="#" onclick="alert('hello');">hello</a>
```

在办公自动化管理系统中，这 3 种方法都有使用到，在后面涉及具体应用时再进行说明。下面先来看两个经常使用到的脚本函数。

（1）验证函数，用于判断表单元素是否为空。如果为空，则返回 false，并将光标焦点定位到出问题的表单元素。程序代码如下：

例程01　代码位置：光盘\TM\03\oa\js\client_js.js

```
//后台登录界面验证脚本
function check(){
    if(login.username.value==""){                  //判断用户名是否为空
        alert("请输入用户名!!");
        login.username.focus();                    //将光标焦点定位到该表单元素
        return false;
```

```
    }
    if(login.pwd.value==""){                    //判断用户密码是否为空
        alert("请输入密码!!");                   //如果为空，弹出提示框
        login.pwd.focus();                      //将光标焦点定位到该表单元素
        return false;                           //返回 false
    }
}
```

（2）选择框函数，用于删除确认。当要执行删除操作时，使用 JavaScript 脚本文件进行确认，以免因为误操作而引起不必要的损失。

例程02 代码位置：光盘\TM\03\oa\js\client_js.js

```
function cfm(){
    if(confirm('确认要删除吗？'))                //选择框函数
        return true;                            //如果选择确认，则返回 true，继续执行
    else
        return false;                           //否则，返回 false
}
```

3.5.2 自定义函数

在 inc 文件夹下，有两个主要文件：chec.php 文件和 func.php 文件。其中，chec.php 文件是权限检查文件，是办公自动化管理系统每个页面都要引用的，func.php 文件是自定义函数文件。

1．chec.php 文件

对办公自动化管理系统来说，用户对每个页面的访问都要有该页的权限才可以，如果没有权限级别的限制，随意访问重要的资源和数据，那么只能说明这是个完全失败的系统。chec.php 文件的代码如下：

例程03 代码位置：光盘\TM\03\oa\inc\chec.php

```
<?php
session_start();                        //开启 session 支持
❶  if(!isset($_SESSION[u_name]))
    echo "<script>alert('您无权访问');location='../index.php';</script>";
❷  if($_SERVER['HTTP_REFERER'] == "")
    echo "<script>alert('本系统不允许从地址栏访问');history.go(-1);</script>";
?>
```

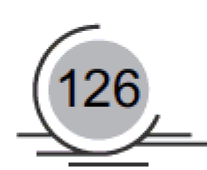
代码贴士

❶ isset()函数：用来判断 session[u_name]是否被创建，如果 session 已经创建，返回 true，否者返回 false。其中 session[u_name]是用户登录验证通过后创建的，如果没有登录的步骤或 session 超时，那么就会提醒用户没有权限。

❷ $_SERVER[HTTP_REFERER]：系统预定义变量，存储的是上一页的 URL 地址。这个判断的作用是防止当前用户访问不属于自己权限内的网页。

2．func.php 文件

限于篇幅，这里只给出两个自定义函数的代码和说明，其他函数请到光盘中自行查看。





（1）读取字段函数 read_field()

该函数的作用是根据传入的 id 值取得相关的字段名，一共有 4 个参数，函数代码如下：

例程04　代码位置：光盘\TM\03\oa\inc\func.php

```
function read_field($conn,$tablename,$fieldname,$n_id){
	$sqlstr = "select ".$fieldname." from ".$tablename." where id = ".$n_id;	//生成 SQL 语句
	$result = mysql_query($sqlstr,$conn);					//执行 SQL 语句
	$rows = mysql_fetch_row($result);					//返回结果集
	return $rows[0];							//返回需要的字段名
}
```

read_field()函数的参数说明如表 3.1 所示。

表 3.1　read_field()函数的参数说明

参 数 名 称	参 数 说 明
$conn	数据库链接资源变量
$fieldname	要查找的字段名
$tablename	要查找的表名
$n_id	要查找的 id 值

（2）处理消息函数 re_message()

该函数的作用是返回消息结果，一共有 2 个参数，函数代码和参数解释如下：

例程05　代码位置：光盘\TM\03\oa\admin\inc\func.php

```
function re_message($result,$l_address){
	if($result)
		echo "<script>alert('操作成功！');location='".$l_address."';</script>"; //如果结果为 true，转到其他页面
	else
		echo "<script>alert('系统繁忙，请稍后再试');history.go(-1);</script>"; //如果结果为 false，回到上一步
}
```

☑ $result：数据库返回结果，如果返回值为 true，说明对数据库的操作成功；如果返回值为 false，说明操作失败。

☑ $l_address：操作成功后，要返回的 URL 地址。

3.6　前台首页设计

对于一个办公自动化管理系统来说，首页不可能，也不允许显示太多的内容和复杂的设计，因为办公自动化管理系统的作用是帮助提高企业的管理效率和职员的工作效率。如果页面过于繁琐，会让职员分不清主次，找不到重点。所以一定要突出重点内容，显示关键功能。

3.6.1　前台首页概述

根据用户对各个功能模块的使用频率和重要程度，本系统的首页中要显示的模块主要有以下 3 部分：

☑ 网站首部导航栏：包括当前用户、所在部门、当前时间、首页、重新登录和退出登录。

☑ 网站左侧导航栏：包括各个管理模块及分类。

  ➢ 企业信息模块：包括公司简介、规章制度、组织结构和企业管理。
  ➢ 企业绩效模块：包括任务绩效、质量绩效和绩效评定。
  ➢ 人事消息模块：包括企业公告、活动安排和消息管理。
  ➢ 审核批示模块：包括发布审核和批示审核。
  ➢ 考勤管理模块：包括上下班登记、病事假登记、加班登记和考勤设置。
  ➢ 个人计划模块：包括工作反馈、周计划、月计划、年计划和任务计划。
  ➢ 职员天地模块：包括职员浏览、意见箱和个人设定。

☑ 网站主显示区：默认显示企业公告、活动安排、个人计划和审核批示。

本案例中提供的前台首页如图 3.16 所示。该首页在本书光盘中的路径为\TM\03\oa\pub_main.php。

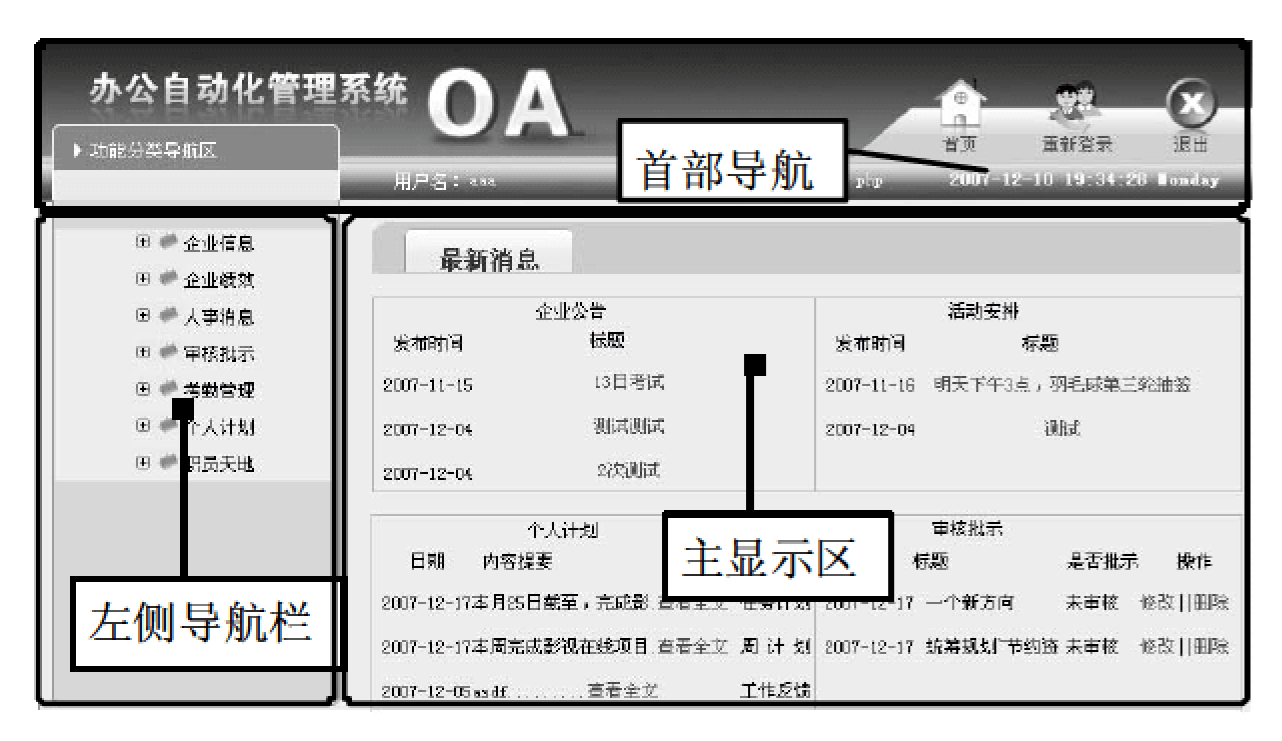


图 3.16　办公自动化管理系统前台首页

3.6.2　前台首页技术分析

目前网页页面的布局主要有两种方式：表格布局和框架布局。本系统前台页面的布局使用的是框架布局。框架布局可以将容器窗口划分为若干个子窗口，每个子窗口可以分别显示不同的网页，网页之间相互独立，没有直接的关联，又由一个网页将这些分开的网页组成一个完整的网页，显示在浏览者的浏览器中。框架布局的好处是：每次浏览者发出对页面的请求时，只下载发生变化的框架页面，其他子页面保持不变。使用框架可以非常方便地完成导航工作。

框架布局的使用格式及属性介绍如下。

1．框架布局格式

框架布局的格式很简单，只要几行代码即可，常用的格式如下：

```
<html>
<head>
…
</head>
<frameset>
    <frame>
    <frame>
</frameset>
<noframes>
```



```
        <body>
        …
        </body>
</noframes>
</html>
```

其中<frameset>和<frame>标签是框架集标记，而<noframes>标签是为了防止浏览器不支持框架而实行的一种补救措施。如果浏览器不支持框架集，就会执行<noframes>标记中的内容，让用户能够正常浏览网页。

2．框架集属性

框架集包含各个框架的信息，通过<frameset>标记来定义。框架是按照行和列来组织的，可以使用<frameset>标记的属性对框架的结构进行设置。下面给出框架集的常用属性值、说明和应用举例，如表 3.2 所示。

表 3.2　框架集的常用属性

属 性 名 称	属 性 说 明	应 用 举 例
COLS	在水平方向上将浏览器分割成多个窗口，取值有 3 种形式：像素、百分比（%）和相对尺寸（*）	<frameset cols="25%,100,*" > <frame></frame> </frameset>
ROWS	在垂直方向上将浏览器分割成多个窗口，取值和 COLS 类似，也是 3 种形式	<frameset rows="25%,100,*" > <frame> <frame> </frameset>
FRAMEBORDER	指定框架周围是否显示边框，取值为 1（显示边框，默认值）或 0（不显示边框）	<framset cols="25%,*" cols="*" frameborder="0"> … </frameset>
FRAMESPACING	指定框架之间的间隔，以像素为单位。默认是无间隔的	<framset cols="25%,*" cols="*" framespacing="1"> … </frameset>
BORDER	指定边框的宽度，frameborder 属性为 1 时该属性才有效	<framset cols="25%,*" cols="*" frameborder="1" border="5"> … </frameset>

3．框架属性

使用<frame>标记可以设置框架的属性，包括框架的名称、框架是否包含滚动条以及在框架中显示的网页等。<frame>标记的常用属性及说明如表 3.3 所示。

表 3.3　框架的常用属性及说明

属 性 名 称	属 性 说 明
NAME	指定框架的名称
SRC	指定在框架中显示的网页文件（包括 HTML、PHP、JSP 等网页文件）
FRAMEBODER	指定框架周围是否显示边框，取值为 1（显示边框，为默认）或 0（不显示边框）

续表

属 性 名 称	属 性 说 明
NORESIZE	可选属性，若指定了该属性，则不能调整框架的大小
SCROLLING	指定框架是否包含滚动条。属性值可以是 yes（有）、no（没有）和 auto（自由）

3.6.3 前台首页的实现过程

前台首页使用的数据表：tb_users、tb_list、tb_person、tb_register、tb_plan

办公自动化管理系统的前台首页采用的是二分栏结构布局。二分栏布局的特点是简洁、大气、个性鲜明，其框架设计依据其内容形式的变化而灵活多变，结构简练，非常符合办公自动化管理系统的风格。本系统中框架布局的具体应用如下。

例程06 代码位置：光盘\TM\03\oa\pub_main.php

```
❶ <frameset rows="111,*" cols="*" frameborder="no" border="0" framespacing="0">
❷     <frame src="top.php" name="topFrame" scrolling="No" noresize="noresize" id="topFrame"
title="topFrame" />
❸ <frameset rows="*" cols="220,*" framespacing="0" frameborder="not" border="0">
❹     <frame src="left.php" name="leftFrame" scrolling="auto" noresize="noresize" id="leftFrame"
title="leftFrame" />
❺     <frame src="main.php" name="mainFrame" scrolling="auto" id="mainFrame" title="mainFrame" />
    </frameset>
</frameset>
❻ <noframes><body>
</body>
</noframes>
```

代码贴士

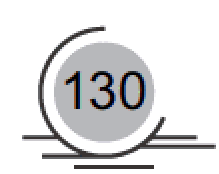
❶ rows="111,*" cols="*"：定义顶部框架。
❷ src="top.php"：定义顶部框架内显示的页面。
❸ rows="*" cols="220,*"：定义下部左侧框架。
❹ src="left.php"：定义左侧框架显示的页面。
❺ src="main.php"：定义中间主框架显示的页面。
❻ <noframes>：当浏览器不支持框架集时显示的内容。

3.7 人事消息模块设计

人事消息模块主要是对文件进行收发管理，模块的设计和实现并不复杂，但在整个系统中的位置却很重要。因为公告栏和意见箱面向的是全体用户，企业的最新动向和职员的意见想法首先都是在这里体现出来的。

3.7.1 人事消息模块概述

人事消息模块主要包含了两部分内容：一部分是面向全体用户的，包括查看公告、活动、意见箱





等；另一部分是仅对人事部开放，其他用户不允许、也不会看到的页面，如公告管理、意见管理等。人事消息用例图如图 3.17 所示。

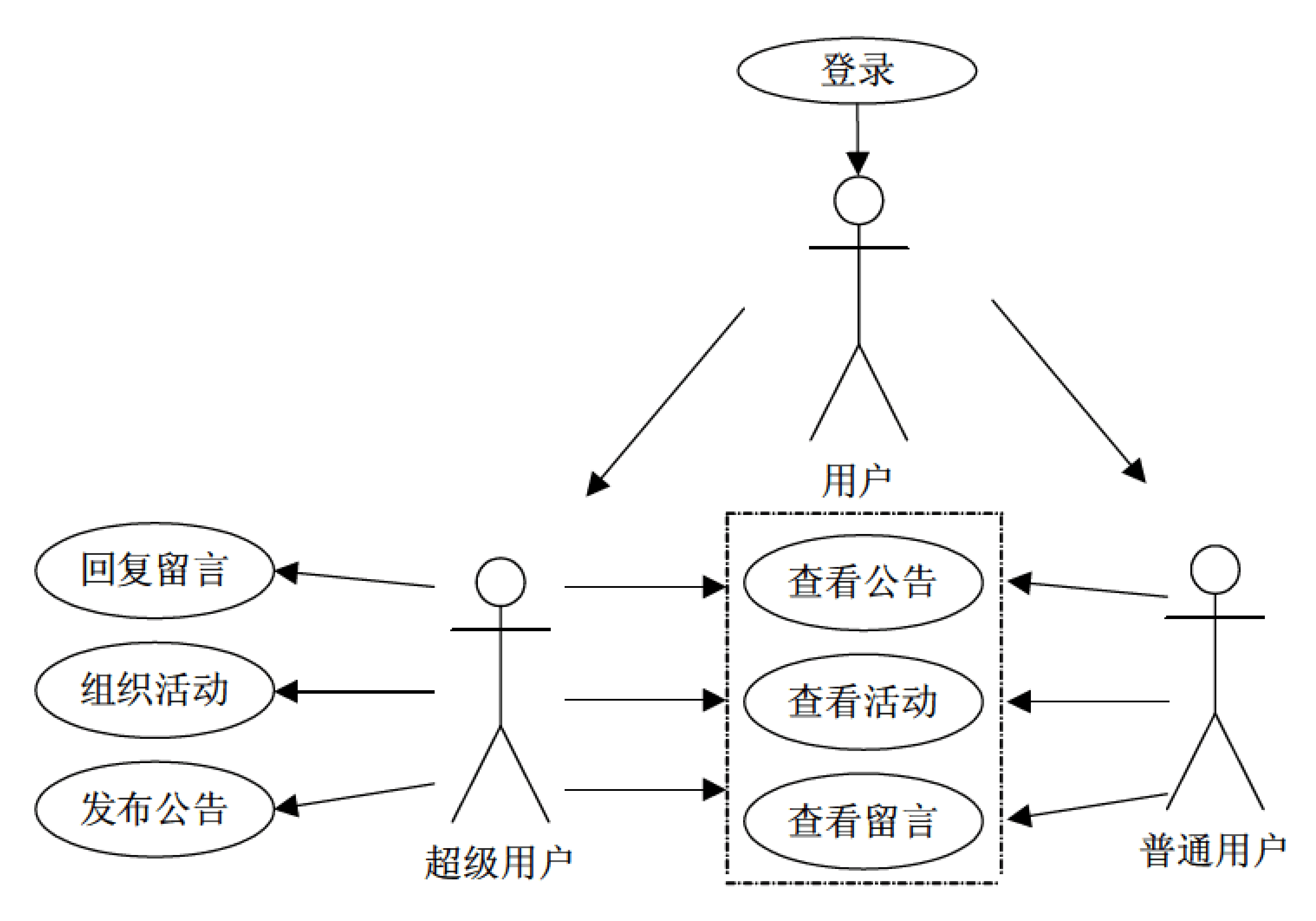


图 3.17　人事消息用例图

3.7.2　人事消息模块技术分析

在人事消息模块中，企业公告和活动安排都是在同一显示页中显示（p_message.php），为了区分不同的操作，需要给显示页（p_message.php）传一个 u_id 值，在数据表中根据不同的 u_id 值，取得不同类别的内容。传值方式有 url 传值方式（如 p_message.php?u_id=1 ）和表单传值方式。

同样地，在接收页中，接收方式也有两种：get 方式和 post 方式。PHP 中使用预定义变量$_GET 和$_POST 来接收传值，格式如下：

```
$_GET/POST[u_id];        //注意，变量名都为大写
```

对于 url 传值方式，接收页始终用$_GET[]变量来接收，如果使用 form 表单传值，那么就要看 form 表单中 method 属性的设置了。如果 method=get，则使用$_GET[]变量；如果 method=post，则使用$_POST[]变量。

PHP 对传值方式和接收方式规定很严格，用$_POST[u_id]接收不了 p_message.php?u_id=1 传过来的参数。同样地，用$_GET[u_id]也接收不着 method=post 的表单元素值。

注意　如果在 php.ini 配置文件中 GLOBAL=ON/OFF 这行的值为 ON，那么直接写$name 就可以调用表单元素的值，而不区分 get 和 post；如果为 OFF，则不可以。直接应用表单名称十分方便，但也存在着安全隐患。推荐关闭 GLOBAL。

3.7.3　消息管理的实现过程

消息管理模块使用的数据表：tb_person

单击“人事消息”选项中的“消息管理”子选项，在主框架（mainFrame）内就会显示消息管理页

面，在页面中显示发布过的消息列表和对消息的操作，包括“修改”、“删除”和“发布新消息”。消息管理页面的运行结果如图 3.18 所示。



图 3.18　消息管理页面运行结果

在本页面中，系统首先通过权限检查文件（chec.php）判断用户是否为登录用户，是否为非法链接。如果被系统判断为非法操作，系统将直接关闭；如果没有异常情况，则系统继续执行下面的代码（在后面的例程中，几乎每个页面都有 chec.php 文件，如果不是特殊说明，则都为正常登录，到时将不再具体说明），当用户通过检查后，系统将访问数据库，从人事消息数据表中（tb_person）读取出消息发布时间、消息标题，显示在页面中，并通过 id 为每条数据设置“修改”和“删除”操作。程序关键代码如下：

例程07　代码位置：光盘\TM\03\oa\rsxx\p_manage.php

```
<?php
	session_start();                                    //开启 session 支持
	include "../inc/chec.php";                          //包含权限检查文件
	include "../conn/conn.php";                         //包含数据库连接文件
?>
<link href="../css/style.css" rel="stylesheet" />       //引入外部 CSS 样式文件
<script src="../js/client_js.js"></script>              //引入 js 文件
…
<?php
/*显示全部消息记录*/
	$sqlstr = "select id,p_time,p_title from tb_person";  //生成 SQL 语句
	$result = mysql_query($sqlstr,$conn);               //执行 SQL 语句，生成结果集
	while($rows = mysql_fetch_row($result)){            //循环输出结果集记录
		echo "<tr>";
		for($i=1;$i<count($rows);$i++){
			echo "<td height=30 style='text-indent: 30px;'>".$rows[$i]."</td>";
		}
		echo "<td><a href='m_message.php?id=".$rows[0]."'>修改</a>/<a
href='d_message_chk.php?id=".$rows[0]."' onclick='return del_mess();'>删除</a></td>";
		echo "</tr>";}
/***************************/
?>
<tr>
	<td height="30" align="right" valign="middle" colspan="3"><a href='add_manage.php' target="mainFrame">
发布新消息</a></td>
</tr>
```



下面分别介绍“发布新消息”、“修改”和“删除”消息的实现过程。

1．发布新消息

单击“发布新消息”超链接，进入发布页面。发布消息页面的运行结果如图 3.19 所示。



图 3.19　发布消息页面运行结果

发布页面中，主要包含一个消息表单元素，部分表单元素如表 3.4 所示。

表 3.4　发布消息页面的主要表单元素

名　称	元 素 类 型	重 要 属 性	含　义
addmess	form	action="add_manage_chk.php" method="post" id="addmess"	发布消息表单
p_title	text	name="p_title" id="p_title"	消息标题
p_content	textarea	name="p_content" id="p_content" cols="60" rows="15"	消息内容
p_type	select	<option value="9">企业公告</option> <option value="10">活动安排</option>	消息类型
submit2	submit	value="发布" onclick="return add_mess();"	“发布”按钮

在消息页面中输入消息的相关内容后，单击“发布”按钮，系统将会跳到消息处理页（add_manage_chk.php）进行处理，将消息存储到数据库中，然后再返回该页。程序关键代码如下：

例程08　代码位置：光盘\TM\03\oa\rsxx\add_manage_chk.php

```
<?php
	session_start();					//开启 session 支持
	include "../inc/chec.php";				//包含权限检查文件
	include "../conn/conn.php";				//包含数据库链接文件
	include "../inc/func.php";				//包含函数处理文件
/*添加新消息*/
	$sqlstr = "insert into tb_person values('','".$_POST[p_title]."','".$_POST[p_content]."',now(),'".$_POST[p_type]."')";
	$result = mysql_query($sqlstr,$conn);
	re_message($result,"p_manage.php");		//消息处理函数
/************/
?>
```

2．修改消息

消息发布后，可以随时对消息内容进行修改。在消息管理页面中，选择要修改的消息列，单击“修改”超链接将进入消息修改页面。在消息修改页面中，有一个和发布页面非常相似的“修改”表单。

消息修改页面中的主要表单元素如表 3.5 所示。

表 3.5　修改页面的表单元素

名　称	元素类型	重要属性	含　义
addmess	form	action="m_message_chk.php" method="post"	修改消息表单
p_title	text	value="<?php echo $rows[1]; ?>"	消息标题
p_content	textarea	<?php echo $rows[2]; ?>	消息内容
p_type	select	<?phpif($rows[4] == "9"){?> <option value="9" selected="selected">企业公告</option> <option value="10">活动安排</option> <?php}else{?> <option value="9">企业公告</option> <option value="10" selected="selected">活动安排</option> <?php}?>	消息类型
id	hidden	value="<?php echo $rows[0]; ?>"	修改消息 id
submit2	submit	value="修改" onclick="return add_mess();"	“修改”按钮

消息修改完成后，单击“修改”按钮，在 m_message_chk.php 页中将会处理修改事件，通过隐藏表单传过来的 id，定位到要修改的消息项，使用 update 语句实现对数据的更新。程序关键代码如下：

例程09　代码位置：光盘\TM\03\oa\rsxx\m_message_chk.php

```
<?php
    session_start();                                    //开启 session 支持
    include "../inc/chec.php";                          //包含权限检查文件
    include "../conn/conn.php";                         //包含数据链接文件
/*更新数据库*/
    $sqlstr = "update tb_person set p_title = '".$_POST[p_title]."',p_content = '".$_POST[p_content]."',p_time = now(),u_id = ".$_POST[p_type]." where id = ".$_POST[id];
    $result = mysql_query($sqlstr,$conn);
    re_message($result,"p_manage.php");                 //消息处理函数
/**************/
?>
```

3．删除消息

每隔一段时间管理员就要定期清理一下无用的消息和活动安排，以加速页面的浏览速度，减轻服务器压力。删除消息的运行结果如图 3.20 所示。



图 3.20　删除消息的运行结果

删除消息时，选择要删除的消息列，单击“删除”超链接，弹出确认删除对话框，单击“确定”按钮，将要删除消息的 id 号传给系统，通过消息处理函数将消息删除。关键代码如下：

例程10　代码位置：光盘\TM\03\oa\rsxx\d_message_chk.php

```
<?php
	session_start();                                        //开启 session 支持
	include "../inc/chec.php";                              //包含权限检查文件
	include "../conn/conn.php";                             //包含数据库链接文件
	include "../inc/func.php";                              //包含函数处理文件
	/*删除数据*/
	$sqlstr = "delete from tb_person where id = ".$_GET[id];
	$result = mysql_query($sqlstr,$conn);
	re_message($result,"./p_manage.php");                   //消息处理函数
	/**********/
?>
```

3.7.4　意见箱的实现过程

意见箱模块使用的数据表：tb_lyb

意见箱是用户匿名发言的一个留言板。对于用户的留言，只有人事部的人员才有权回复或删除，对于其他部门的员工则只可以留言和查看，而无法进行其他操作。意见箱的运行结果如图 3.21 所示。

在本页面中，系统首先读取出意见箱数据表（tb_lyb）中的所有数据，并逐条显示，在每条显示记录后，有一个判断式，判断用户所属部门是否为人事部，如果是人事部成员，将显示“回复”和“删除”超链接；如果不是人事部，则不显示这两个操作。这里是强制性的，也就是说，即使是管理员，也无法设置这里的权限，只有开发人员才可以改变。如果想使用这种技术，就一定要确保这项功能权限不会轻易地更改。意见箱页面的关键代码如下：



图 3.21　意见箱页面的运行结果

例程11　代码位置：光盘\TM\03\oa\zytd\lyb.php

```
<?php
session_start();                                        //开启 session 支持
include "../conn/conn.php";                             //包含数据库链接文件
include "../inc/chec.php";                              //包含权限检查文件
include "../inc/func.php";                              //包含自定义函数文件
/*显示意见箱信息*/
$l_sql = "select id,l_title,l_content,l_time,is_reply,r_back from tb_lyb order by id desc";
$l_rst = mysql_query($l_sql,$conn);
…
	while($l_rows = mysql_fetch_array($l_rst)){         //开始循环输出每条记录
?>
  <tr>
	<td width="15%" height="30" align="left" valign="middle" scope="col">
```

```
        <?php echo $l_rows[l_time]; ?>
        </td>
        <td align="left" valign="middle" scope="col">
        主题：<?php echo $l_rows[l_title]; ?>
        </td>
        <!-- 判断登录用户是否为人事部成员-->
       <td width="15%" align="center" valign="middle" scope="col">
        <?php if(($_SESSION[u_depart]) == "人事部"){ ?>
        <!--当登录用户是人事部成员时，显示可执行操作 -->
        <a href="t_back.php?id=<?php echo $l_rows[id]; ?>">回复</a>  <a href="del_ly_chk.php?id=
<?php echo $l_rows[id]; ?>" onclick="return del_mess()">删除</a>
        <?php }
         ?> 
        </td>
        <!-- ---------------------------------------- -- >
    </tr><tr> <td height="75" colspan="3" align="left" valign="middle" scope="col" >
        <?php echo $l_rows[l_content]; ?>
        </td></tr>
<?php
        <!--当字段 is_reply 的值为 1 时，说明有回复信息-->
        if($l_rows[is_reply] == 1){
?>
    <tr>
        <td height="25" colspan="3" align="left" valign="middle" headers="50" scope="col"
onmouseover="this.style.backgroundColor='#FFEEBC'" onmouseout="this.style.backgroundColor=''">
        <font color="#FF0000">
        <!--显示回复信息-->
        <?php   echo $l_rows[r_back]; ?>
        </font></td> </tr>
<?php}}
/***************/
?></table>
<!--留言菜单-->
        ...
<!-- ------------------- -- >
```

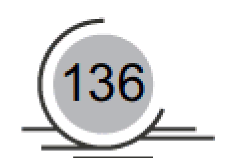
下面介绍匿名留言和回复、删除留言的实现过程。

1. 匿名留言

留言页面主要是一个 HTML 表单，因为允许匿名留言，所以表单中没有隐藏域，该页面中的表单元素列表如表 3.6 所示。

表 3.6 留言页面的菜单表单元素表

名 称	元素类型	重要属性	含 义
lyb	form	id="lyb" name="lyb" method="post" action="lyb_chk.php"	留言表单
l_title	text	type="text" name="l_title" size="30"	意见主题
l_content	textarea	name="l_content" cols="75" rows="10"	意见内容
Submit1	hidden	type="reset" value="重置"	“重置”按钮
Submit	submit	type="submit" value="提交" onclick="return add_lyb();"	“提交”按钮





当用户填写完意见标题和意见内容后，单击“提交”按钮，发布意见，系统将转到意见处理页（lyb_chk.php）中对意见进行处理：如果意见标题和意见内容都不为空，则执行 insert 语句，向表 tb_lyb 中添加一条新数据。程序关键代码如下：

```
<?php
	session_start();                                          //开启 session 支持
	include "../conn/conn.php";                               //包含数据库链接文件
	include "../inc/chec.php";                                //包含权限检查文件
	include "../inc/func.php";                                //包含自定义函数文件
/*判断输入是否为空，如果为空，则返回；如果无错误，则执行 insert 语句添加新记录*/
if((trim($_POST[l_title]) != "") and (trim($_POST[l_content]) != "")){
		$l_sql = "insert into tb_lyb values('','".$_POST[l_title]."','".$_POST[l_content]."',now(),'','')";
		$l_rst = mysql_query($l_sql,$conn);
		re_message($l_rst,"lyb.php?u_id=24");             //使用自定义函数
/*****************************************************************************/
}
else
		echo "<script>alert('内容和消息不允许为空');history.go(-1);</script>";
?>
```

2．回复、删除留言操作

在意见箱页面中，当系统判断当前登录用户为人事部成员后，将显示留言信息后面的、可操作的功能模块“回复”和“删除”。单击相应留言主题后面的“回复”超链接，进入回复留言页面。回复留言页面的运行结果如图 3.22 所示。

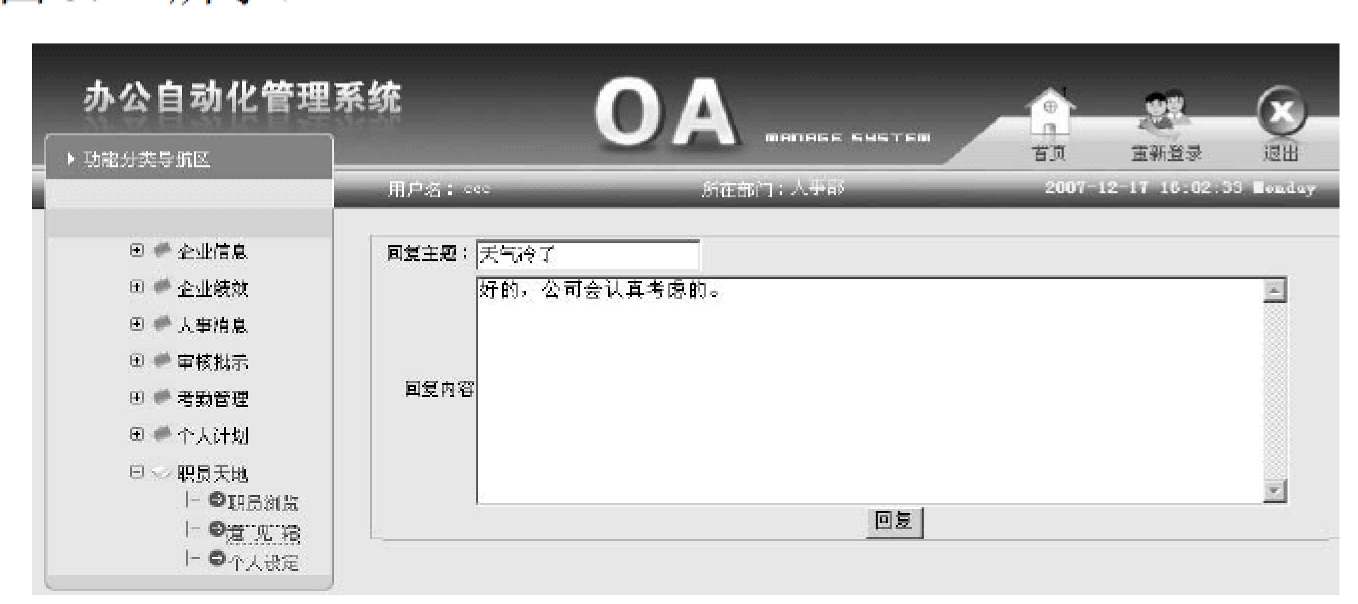


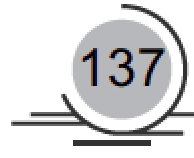
图 3.22　回复留言页面的运行结果

回复留言页面中的主要表单元素如表 3.7 所示。

表 3.7　回复留言表单的主要元素

名　称	元素类型	重要属性	含　义
tback	form	action="t_back_chk.php" method="post"	回复留言表单
t_title	text	value="<?php echo read_field($conn,"tb_lyb","l_title",$_GET[id]); ?>"	意见主题
r_back	textarea	name="r_back" id="r_back" cols="75" rows="10"	回复内容
id	hidden	name="id" value="<?php echo $_GET[id]; ?>"	回复消息 id
submit2	submit	value="回复" onclick="return re_back();"	“回复”按钮

在回复留言页面中，当用户输入回复信息后，单击“回复”按钮提交表单，在表单处理页（t_back_



chk.php）中系统将会根据回复信息，更新表记录。关键代码如下：

例程12　代码位置：光盘\TM\03\oa\zytd\ t_back_chk.php

```
<?php
session_start();                                //开启 session 支持
include "../conn/conn.php";                     //包含数据库链接文件
include "../inc/chec.php";                      //包含权限检查文件
include "../inc/func.php";                      //包含自定义函数文件
/*根据 id，更新数据库*/
$sqlstr = "update tb_lyb set r_back='".$_POST[r_back]."',is_reply = 1 where id = ".$_POST[id];
$result = mysql_query($sqlstr,$conn);
re_message($result,"lyb.php?u_id=24");   //信息函数
/********************/
?>
```

对于不需要再保留的留言信息，可将其删除。单击相应消息主题后面的“删除”超链接，弹出确认删除提示对话框，单击“确定”按钮将消息的 id 传给系统，通过 SQL 语句将其删除。程序关键代码如下：

例程13　代码位置：光盘\TM\03\oa\zytd\del_ly_chk.php

```
<?php
session_start();                                              //开启 session 支持
include "../conn/conn.php";                                   //包含数据库链接文件
include "../inc/chec.php";                                    //包含权限检查文件
include "../inc/func.php";                                    //包含自定义函数文件
$sqlstr = "delete from tb_lyb where id = ".$_GET[id];         //删除信息的 SQL 语句
$result = mysql_query($sqlstr,$conn);                         //执行语句
re_message($result,"lyb.php?u_id=24");                        //返回结果
?>
```

3.8　考勤管理模块设计

考勤功能是办公自动化管理系统中每天都要使用到的功能模块之一，也是比较重要的模块之一。除了包括正常的上下班登记功能外，还包括病事假登记和加班登记等特殊登记功能。管理员通过考勤记录实现对上下班及加班标准时间的设置。

3.8.1　考勤管理模块概述

考勤模块主要的功能介绍如下。

☑　上下班登记：包括上班登记和下班登记。
☑　病事假登记：包括病假登记和事假登记。
☑　加班登记：包括加班上班登记和加班下班登记。
☑　考勤记录：包括上下班标准时间设置和加班标准时间设置。

考勤模块的活动图如图 3.23 所示。



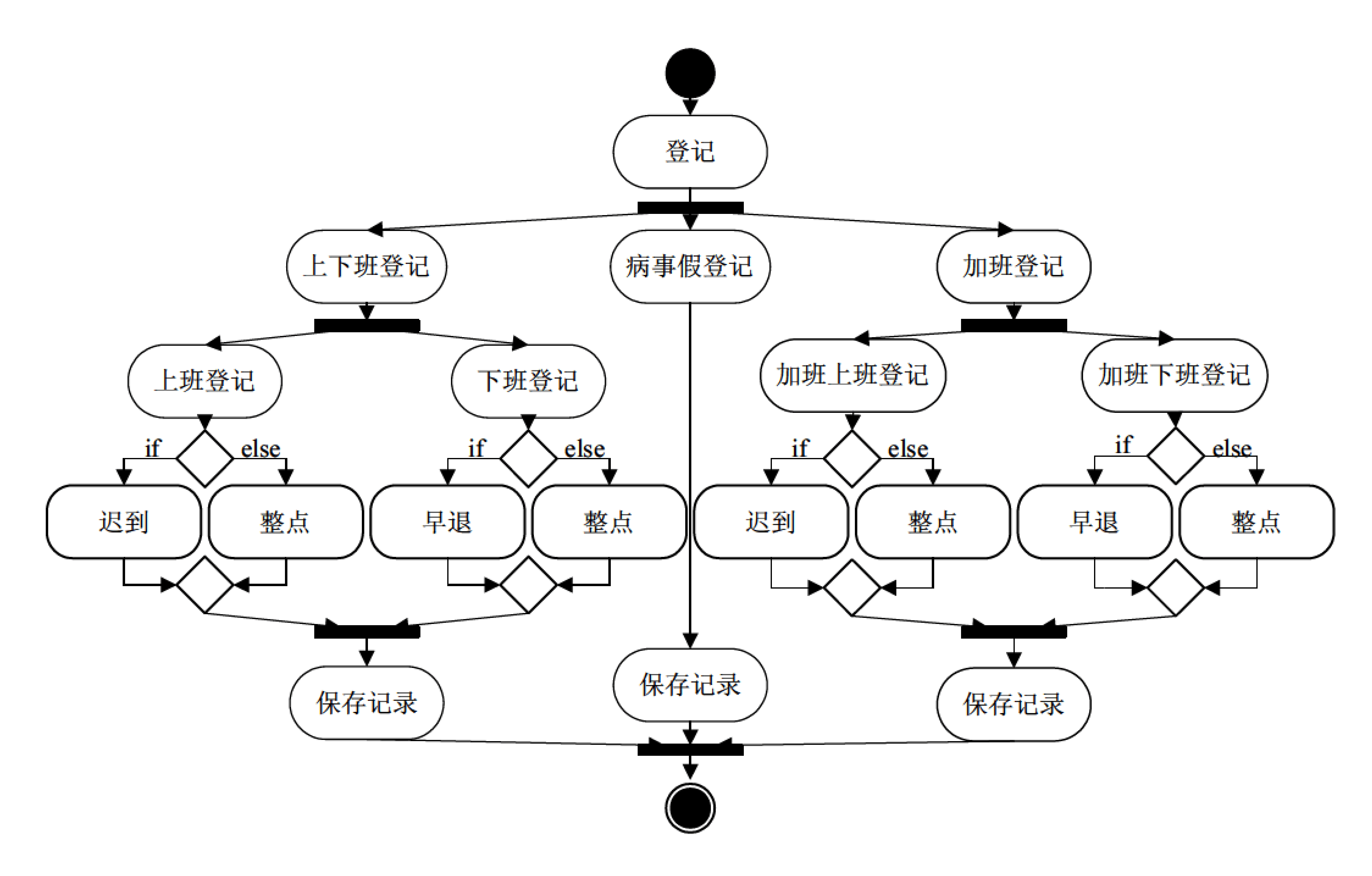


图 3.23　考勤管理活动图

3.8.2　考勤管理模块技术分析

在 PHP 中查询数据表中数据的一般步骤如下：

（1）链接数据库（链接文件 conn.php）。

（2）执行 SQL 语句，返回查询结果集（$result = mysql_query($sqlstr,$conn)。

（3）配合 while 循环语句输出查询结果（while($rows = mysql_fetch_row($result))）。

（4）关闭数据库。

在步骤（3）中，如果要取得某个字段的值，使用$rows[$num]即可。但通过图 3.23，读者可以发现，从开始登记到保存登记记录，虽然只是一个功能的实现，却要进行至少 3 次判断。如果都用“0,1,2…”做数组下标，过一段时间后，根本记不起取得的数据是什么、有什么作用。在后期测试和系统维护时，还要逐一地查找数据表来比对。除了 mysql_fetch_row()函数外，PHP 还提供了其他相似的函数，下面就来学习一下。

1．mysql_fetch_array()函数

mysql_fetch_array()函数的用法和 mysql_fetch_row()函数十分相似，唯一不同的是结果集的数组下标是所查找的数据表的字段值。

例如，读取数据表 tb_register 中的“登记时间”字段，代码如下：

```
$str = $rows[r_time];
```

这样，即使时间间隔比较长，甚至换人开发，也能明白该变量的含义。

2．mysql_fetch_object()函数

mysql_fetch_object()函数与其他两个函数的最大不同就在于：获取的结果集是以对象的形式存储的，而不是数组的形式。也就是说，在读取字段变量时所使用的格式不再是$rows[$num]，代码如下：

```
$str = $rows-->r_time;
```

下面来看一个实例，分别使用上面 3 种不同的函数来获取数据表 tb_setup 的 id 值，代码如下：

```
<?php
    /*链接数据库*/
    $conn = mysql_connect("localhost","root","root");
    mysql_select_db("db_office",$conn);
    mysql_query("set names gb2312");
    /*创建和执行 SQL 语句*/
    $sqlstr = "select * from tb_setup";
    $result = mysql_query($sqlstr,$conn);
    /*使用 mysql_fetch_row()函数输出字段值*/
    $rows_1 = mysql_fetch_row($result);
    $str = "mysql_fetch_rows 函数：".$rows_1[0]."<br>";
    /*使用 mysql_fetch_array()函数输出字段值*/
    $rows_2 = mysql_fetch_array($result);
    $str .="mysql_fetch_array 函数：".$rows_2[id]."<br>";
    /*使用 mysql_retch_object()函数输出字段值*/
    $rows_3 = mysql_fetch_object($result);
    $str .= "mysql_fetch_object 函数：".$rows_3->id."<br>";
    echo "<script>alert('".$str."');</script>";
?>
```

输出结果为：

```
mysql_fetch_rows 函数：1
mysql_fetch_array 函数：2
mysql_fetch_object 函数：3
```

3.8.3　上下班登记的实现过程

上下班登录使用的数据表：tb_register

单击“考勤管理”选项中的“上下班登记”子选项，在主显示区（mainFrame）内显示个人上下班的登记记录和登记链接，用户可以在这里进行登记和查找。上下班登记页面的运行结果如图 3.24 所示。



图 3.24　上下班登记的运行结果

在上下班登记页面中，通过登录用户 id，从数据表 tb_register 中返回用户的登记记录集，并显示在页面中。在显示数据时，需要进行 2 次判断：一次判断用户的登记类型（上班还是下班）；另一次判断用户的登记状态（迟到、早退、正点上下班）。关键代码如下：





例程14　代码位置：光盘\TM\03\oa\kqgl\work_note.php

```
<?php
	session_start();						//开启 session 支持
	include "../inc/chec.php";				//包含权限检查文件
	include "../conn/conn.php";				//包含数据库链接文件
	include "../inc/func.php";				//包含自定义函数文件
?>
<link href="../css/style.css" rel="stylesheet" />		//引进外部 css 样式文件
<script src="../js/client_js.js"></script>			//引进 js 脚本文件
…
❶	<a href="#" onclick="javascript:openWin=open('p_login.php?r_id=<?php echo $_GET[u_ id]; ?>' ,'',
'width= 420,height=260,scrollbars=no');">
<?php
	/*判断是哪种功能类型*/
❷	if($_GET[u_id] == 14){
		echo "上下班登记";
	}else if($_GET[u_id] == 15){
		echo "病事假登记";
	}else if($_GET[u_id] == 16){
		echo "加班登记";
	}
?></a>
<?php
	/*显示登记信息*/
	$sqlstr = "select id,r_date,r_time,r_type,r_state,r_remark from tb_register where r_id = ".$_GET[u_id]."
and p_id = ".$_SESSION[id];
	$result = mysql_query($sqlstr,$conn);
	while($rows = mysql_fetch_array($result)){
?>
	<tr>
	<!--输出登记日期-->
		<td height="25" align="center" valign="middle"><?php echo $rows[r_date]; ?></td>
	<!--输出登记时间-->
		<td height="25" align="center" valign="middle"><?php echo $rows[r_time]; ?></td>
		<td height="25" align="center" valign="middle">
<!--输出登记类型-->
		<?php
❸			switch($rows[r_type]){
				case 0:
					echo  "下班";
					break;
				case 1:
					echo "上班";
					break;
				case 2:
					echo "加班签到";
					break;
				case 3:
					echo "加班签退";
					break;
```

```
                case 4:
                    echo "病假";
                    break;
                case 5:
                    echo "事假";
                    break;
            }?></td>
        <!-- -------------------- -->
        <td height="25" align="center" valign="middle">
        <!--输出登记状态-->
        <?php
❹           switch($rows[r_type]){
                case 1:
                    echo ($rows[r_state] == 0)?"正点上班":"迟到";
                    break;
                case 0:
                    echo ($rows[r_state] == 0)?"早退":"正点下班";
                   break;
                case 2:
                    echo ($rows[r_state] == 0)?"正点加班":"晚点加班";
                    break;
                case 3:
                    echo ($rows[r_state] == 0)?"加班早退":"加班下班";
                   break;
                case 4:
                    echo ($rows[r_state] == 0)?"病假":"事假";
                    break;
            }?>
    <! -- ---------------------------------- -- >
        </td>
        <!--输出备注-->
        <td height="25" align="center" valign="middle"><?php echo ($rows[r_remark] != null)?$rows
[r_remark]:无; ?></td>
    </tr>
<?php    }?>
/******************************/
...
```

代码贴士

❶ onclick="…"：这里的 js 脚本使用的是第三种方式，在表单元素或标签中使用。当单击超链接时，将触发一个窗口事件，打开一个新的窗口。open()函数的基本格式为 open(url,name,feature)。其中 url 为要打开的地址，name 为打开窗体的名字，features 为窗体属性特征，如宽、高等。

❷ 根据 u_id，也就是参数 u_id 的值来判断是哪种功能类型。分 3 种情况：上下班登记、病事假登记和加班登记。

❸ 判断登记类型，一共分 6 种情况：上班、下班、病假、事假、加班签到和加班签退。

❹ 判断状态类型，一共分 10 种情况：上班迟到、上班早退、正点上班、正点下班、正点加班、加班下班、加班晚点、加班早退、病假和事假。

当单击“上下班登记”超链接时，就会弹出登记页面，用户可以进行上下班的登记操作，在“登记类型”下拉列表框中选择登记类型，如果是迟到或早退等特殊情况，需要在“备注”列表框中说明

原因，最后单击“登记”按钮进行登记，如果登记成功，将显示提示对话框。登记页面的运行结果如图 3.25 所示。



图 3.25　登记页面的运行结果

在登记界面中，主要有 3 个表单元素和一个隐藏表单，如表 3.8 所示。

表 3.8　上下班登记的表单元素

名　　称	元 素 类 型	重 要 属 性	含　　义
Pl	form	method="post" action="p_login_chk.php"	登记表单
u_name	text	value="<?php echo $_SESSION[u_name]; ?>" readonly="readonly"	登记人
u_type	select	<?php if(**$_GET[r_id] == 14**){?> <option value="1">上班</option> <option value="0">下班</option> <?php }else if(**$_GET[r_id] == 15**){?> <option value="4">病假</option> <option value="5">事假</option> <?php }else if(**$_GET[r_id] == 16**){?> <option value="2">加班签到</option> <option value="3">加班签退</option> <?php }?>	登记类型
r_remark	textarea	name="r_remark" rows="5"	备注
r_id	hidden	value="<?php echo $_GET[r_id]; ?>"	功能 id
Submit2	submit	value="回复" onclick="return re_back();"	“登记”按钮

选择相应的登记类型，如上班，提交表单后，系统就会跳到处理页（p_login_chk.php）处理。在处理页中，首先判断登记类型（上班、下班），然后根据登记类型取得相对应的标准时间，通过两个时间的对比，得到登记状态（迟到、早退、正点上下班），最后执行 insert 语句，添加新记录。程序关键代码如下：

例程15　代码位置：光盘\TM\03\oa\kqgl\p_login_chk.php

```
<?php
session_start();                                         //开启 session 支持
include "../conn/conn.php";                              //包含数据库链接文件
include "../inc/chec.php";                               //包含权限检查文件
/*1 代表迟到，0 代表正点登记*/
/*判断功能类型，如上下班登记、加班登记*/
if(($r_id == "14") or ($r_id == "16")){
/*判断登记类型，如上班、下班等*/
      if($u_type == 0)
```

```
            $t_sql = "select * from tb_setup where id = 2";         //取得下班标准时间
        else if($u_type == 1)
            $t_sql = "select * from tb_setup where id = 1";         //取得上班标准时间
        else if($u_type == 2)
            $t_sql = "select * from tb_setup where id = 3";         //取得加班上班时间
        else if($u_type == 3)
            $t_sql = "select * from tb_setup where id = 4";         //取得加班下班时间
/*************************************************************/
        $t_rst = mysql_query($t_sql,$conn);
        $t_rows = mysql_fetch_row($t_rst);
        $s_time = $t_rows[2];                                       //规定的标准时间
❶      $now_time = date("H:i:s");                                  //当前的登记时间
❷      $l_sql = "insert into tb_register (r_date,r_time,r_type,r_state,r_remark,r_id,p_id)
values('".date("Y-m-d")."','".date("H:i:s")."','".$_POST[u_type]."','".((strtotime($now_time) - strtotime($s_time) >
0)?1:0)."','".$_POST[r_remark]."','".$_POST[r_id]."','".$_SESSION[id]."')";
}else if($r_id == "15")
        $l_sql = "insert into tb_register (r_date,r_time,r_type,r_state,r_remark,r_id,p_id)
values('".date("Y-m-d")."','".date("H:i:s")."','".$_POST[u_type]."','','".$_POST[r_remark]."','".$_POST[r_id]."','".$_SE
SSION[id]."')";
$l_rst = mysql_query($l_sql,$conn);
/*登记完成后，关闭窗口*/
if($l_rst)
        echo "<script>alert('登记完成');window.close();</script>";
else
        echo "<script>alert('错误');history.go(-1);</script>";
        /**********************/
?>
```

代码贴士

❶ date()时间函数可以通过不同的组合返回不同的时间样式。例如程序中的 date("H:i:s")，还可以加上年、月、日、星期等，如 date("Y-m-d H:i:s l")。

❷ ((strtotime($now_time) - strtotime($s_time) > 0)?1:0)：根据以上内容得到的 SQL 语句，其中 strtotime 函数是 PHP 的一个时间函数，作用是将一个日期时间描述解析为一个 unix 时间戳，如 strtotime("2007-07-11 13:49:29") 输出的结果就是 1184132969，这时对两个时间进行计算，就可以得到时间的大小关系了。

3.8.4 设置时间的实现过程

考勤管理使用的数据表：tb_register、tb_setup

考勤记录页面的功能主要有两部分：一是显示所有员工的考勤信息；二是设置标准时间。显示所有员工的考勤信息的实现过程和 3.8.3 节中的实现方式相类似，唯一不同的就是包括了全体职员，只要略微改动 SQL 语句即可。

下面主要来看一下设置标准时间页面。在“考勤记录”页面中单击“设置时间”超链接，进入设置时间页面，如图 3.26 所示。

图 3.26　设置时间页面

在本页面中，从 tb_setup 表中读取预定义的标准时间，有 4 个标准时间，分别是“上班签到”、“下班签退”、“加班签到”和“加班签退”。程序关键代码如下：



例程16　代码位置：光盘\TM\03\oa\kqgl\set.time.php

```
<?php
    session_start();                                           //开启 session 支持
    include "../inc/chec.php";                                 //包含权限检测文件
    include "../conn/conn.php";                                //包含数据库链接文件
?>
<link href="../css/style.css" rel="stylesheet" />              //引入 css 外部链接文件
<script src="../js/client_js.js"></script>                     //引入 js 脚本文件
<form name="form1" method="post" action="set_time_chk.php">
<table border="1" cellspacing="0" cellpadding="0" background="../images/bg.jpg">
  <tr>
    <td width="150" height="25" align="center" valign="middle">上班签到</td>
    <td width="150" height="25" align="center" valign="middle">下班签退</td>
    <td width="150" height="25" align="center" valign="middle">加班签到</td>
     <td width="150" height="25" align="center" valign="middle">加班签退</td>
  </tr>
<?php
    $sqlstr = "select * from tb_setup";                        //提取表中的标准时间
    $result = mysql_query($sqlstr,$conn);                      //返回结果集
    $num = 0;                                                  //设置序号
?>
  <tr>
<?php while($rows = mysql_fetch_row($result)){ ?>
    <td height="30" align="center" valign="middle">
❶    <input type="text" name="l_time<?php echo $num;?>" value="<?php echo $rows[2]; ?>" size=15></td>
<?php $num++;}?>
  </tr> <tr>
❷    <td height="30" align="center" valign="middle"><input type="submit" name="u_logo" value="设置"></td>
    <td height="30" align="center" valign="middle"><input type="submit" name="d_logo" value="设置"></td>
    <td height="30" align="center" valign="middle"><input type="submit" name="a_logo" value="设置"></td>
    <td height="30" align="center" valign="middle"><input type="submit" name="q_logo" value="设置"></td>
  </tr>
</table>
</form>
```

代码贴士

❶ 变量$num 是为了在 while 循环语句中输出不同的表单元素的名称，每次循环 $num 自加 1。

❷ 生成 4 个设置时间的按钮，在处理页中，根据不同的名字做不同的处理。

当调整了其中的一项后，单击“设置”按钮，进入到处理页，在处理页中，会根据按钮的名称和 text 元素的 value 值来进行相关的操作。程序关键代码如下：

例程17　代码位置：光盘\TM\03\oa\kqgl\set.time_chk.php

```
<?php
    session_start();                                           //开启 session 支持
    include "../inc/chec.php";                                 //包含权限检查文件
    include "../conn/conn.php";                                //包含数据库链接文件
    /*根据名称判断事件是哪个按钮产生的，并生成不同的 SQL 语句*/
    if(isset($_POST[u_logo])){
        $sqlstr = "update tb_setup set l_time ='".$_POST[l_time0]. "' where id = 1";     //设置上班时间
```

```
    }else if(isset($_POST[d_logo])){
        $sqlstr = "update tb_setup set l_time ='".$_POST[l_time1]. "' where id = 2";       //设置下班时间
    }else if(isset($_POST[a_logo])){
        $sqlstr = "update tb_setup set l_time ='".$_POST[l_time2]. "' where id = 3";       //设置加班上班时间
    }else if(isset($_POST[q_logo])){
        $sqlstr = "update tb_setup set l_time ='".$_POST[l_time3]. "' where id = 4";       //设置加班下班时间
    }
    else
        echo "<script>alert('非法登录');window.close();</script>";
    /*****************************************************/
    $result = mysql_query($sqlstr,$conn);
    if($result)
        echo "<script>alert('设置成功');window.close();</script>";
    else
        echo "<script>alert('设置失败');window.close();</script>";
?>
```

3.9 后台首页设计

由于后台管理主要是针对管理员开放的，普通用户无法访问到这里，所以在页面风格的设计上相对要简单得多。不需要艳丽的图片、极炫的特效，只要干净、整洁，给人一种朴实无华的感觉是最好的。

3.9.1 后台首页概述

如果说前台是系统的外衣，那么后台就是系统的灵魂了。数据更新、系统升级，甚至是安全隐患，都源于后台的设计是否完善，功能是否强大。可以说，后台越完美，系统拥有的外衣才会越华丽，人们的目光才会被吸引。

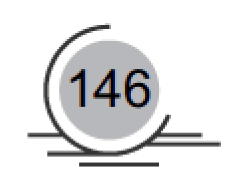
办公自动化管理系统的后台首页主要包括的功能如下。

☑ 网站首部导航栏：包括当前时间、首页、重新登录和退出登录。
☑ 网站左侧导航栏：包括各个管理模块及分类。
☑ 部门管理模块：主要用于对部门信息的操作，包括添加、修改、删除和查看功能。
☑ 职员管理模块：主要用于对职员信息的统一管理，包括添加、修改、删除和查找职员。
☑ 权限管理模块：主要用于对部门、职员和用户组的功能权限分配。
☑ 系统管理模块：维护系统日志、修改管理员密码和系统数据的备份。

本案例中提供的后台首页如图 3.27 所示。该首页在本书光盘中的路径为\TM\03\admin\admin_main.php。



图 3.27　办公自动化管理系统后台首页



3.9.2 后台首页技术分析

后台页面和前台页面的布局是相似的，都是采用“上方固定，左侧嵌套”的框架结构。在上方框架（topFrame）显示 Banner 和首部导航，在左侧框架（leftFrame）中显示功能列表。中间的显示区（manFrame）显示各级窗体。

3.9.3 后台首页的实现过程

后台首页使用的数据表：tb_controller

下面来看一下后台页面中框架布局的应用，程序代码如下：

例程18 代码位置：光盘\TM\03\oa\admin\admin_main.php

```
<!--定义顶部框架和框架内显示的内容 -->
<frameset rows="88,*" cols="*" frameborder="no" border="1" framespacing="0">
  <frame src="top.php" name="topFrame" scrolling="No" noresize="noresize" id="topFrame" title="topFrame" />
  <!--定义嵌套的左侧框架和框架内显示的内容-->
  <frameset rows="*" cols="216,*" framespacing="0" frameborder="no" border="1">
    <frame src="left.php" name="leftFrame" scrolling="No" noresize="noresize" id="leftFrame" title="leftFrame" />
    <!--定义主框架内容 -->
    <frame src="main.php" name="mainFrame" id="mainFrame" title="mainFrame" />
  </frameset>
</frameset>
<!--当浏览器不支持框架时所显示的内容-->
<noframes><body>
</body>
</noframes></html>
```

3.10 部门管理模块设计

一个成熟、稳定的企业一定要有一个合理的部门结构，包括每个部门需要做什么工作、部门之间的协调关系和部门之间的上下级关系等，都要明确下来，这样才能各尽其职，不会出现资源浪费，提高企业整体的效率。

3.10.1 部门管理模块概述

部门管理模块包括查看部门和添加部门两大部分，下面分别介绍。

1. 查看部门

查看各个部门名称和部门之间的上下级关系，还可以对部门的信息和级别进行修改，如果是最底层的部门，则可以对其进行删除操作。

2．添加部门

可以添加新的部门，包括部门名称、上级部门和部门备注。

部门管理框架如图 3.28 所示。

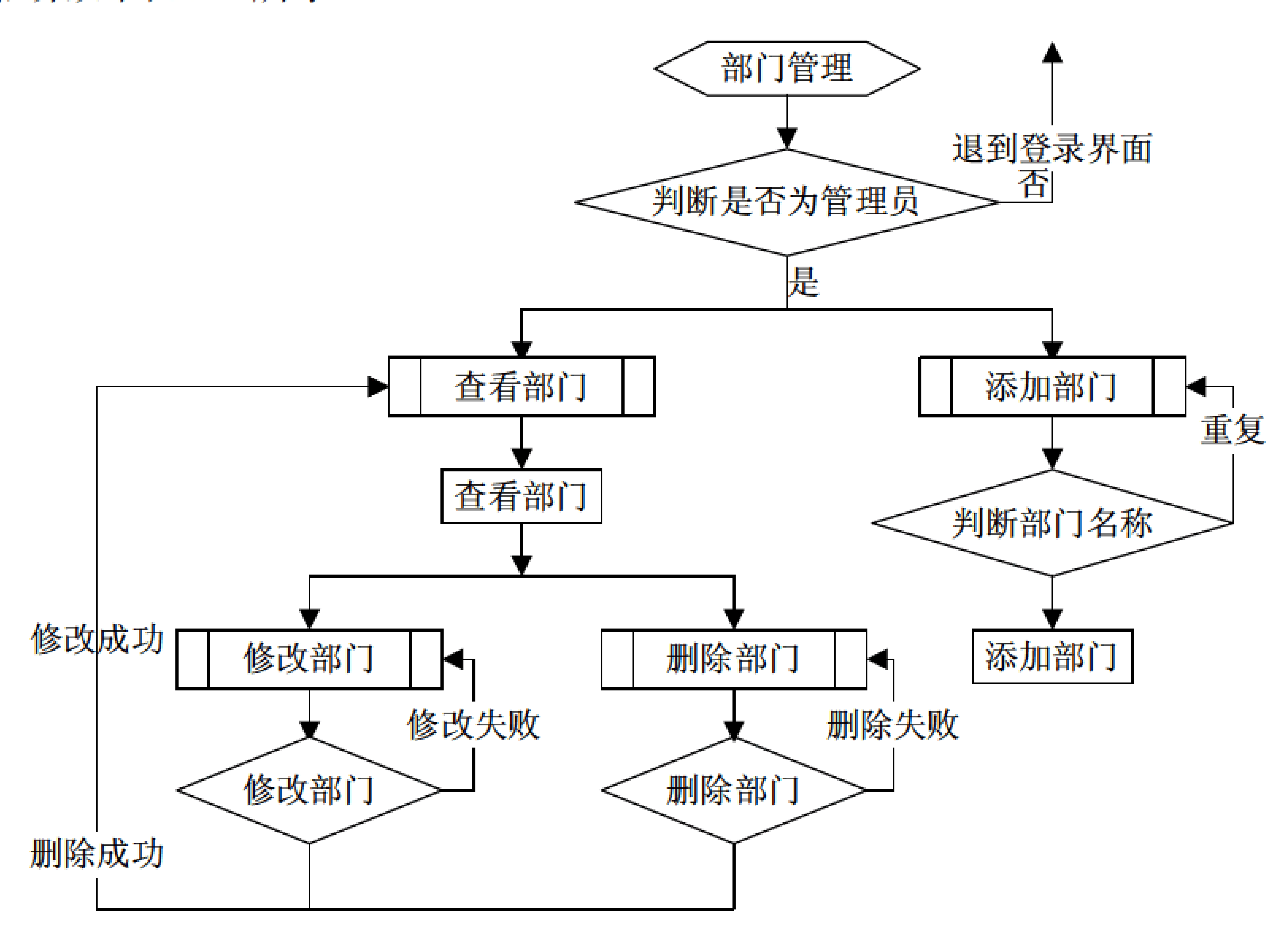


图 3.28　部门管理操作流程图

3.10.2　部门管理模块技术分析

部门管理中的技术难点是：如何实现查看部门模块中的动态显示部门结构的功能。这里用 JavaScript+递归函数来实现。下面详细讲解。

当用户单击“查看部门”时，系统开始查找 tb_depart 数据库。首先找到根部门的信息列表，并判断第一个根部门是否有下级部门。如果没有，则直接输出部门名称、修改和删除操作，然后查找下一个根部门；如果有下级部门，则调用自定义函数 list_menu($num,$wid,&$m)。其中：

☑　$num：为上级部门 id 号。

☑　$wid：为表格宽度。

☑　$m：为表单元素 id 变量值。

在这个函数中，执行步骤如下：

（1）查找上级部门为$num 的部门列表。

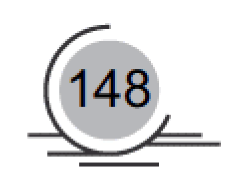
（2）判断第一个部门是否有下级部门。

（3）如果没有，则输出部门名称、修改和删除操作，然后查找下一个同级部门。

（4）如果有下级部门，则再次调用 list_menu($num,$wid,&$m)函数。

（5）重复步骤（1）～（4）。

这时，相信读者已经看出：list_menu()正是一个递归函数。递归函数就是在当满足一定的条件时，函数自己调用自己，经过层层调用，最终达到解决问题的目的。list_menu()函数的代码如下：



例程19　代码位置：光盘\TM\03\oa\admin\inc\func.php

```
<?php
	function list_menu($num,$wid,&$m){
	/*连接数据库*/
	$conn = mysql_connect("localhost","root","root");
	mysql_select_db("db_office",$conn);
	mysql_query("set names gb2312");
	/*****************/
	/*查询同级部门*/
	$sqlstr = "select * from tb_depart where up_depart = ".$num;
	$result = mysql_query($sqlstr,$conn);
	/*******************/
?>
	<!--隐藏域-->
<tr id="OpenMe<?php echo $m; ?>" style="display:none;">
<td> <table width="<?php echo $wid; ?>%"  border="0" cellspacing="0" cellpadding="0">
	<tr>
	<td height="25" align="center"><table width="<?php echo ($wid -3); ?>%" border="0" cellspacing="0"
cellpadding="0">
<?php
/*循环输出同级部门*/
	while($rows = mysql_fetch_row($result)){
	/*查看下属部门*/
		$sqlstr1 = "select * from tb_depart where up_depart = ".$rows[0];
		$result1 = mysql_query($sqlstr1,$conn);
		$nu = mysql_num_rows($result1);
	/*当前部门没有下属部门时*/
		if(!$nu){
			$m += 1;
?>
❶	<tr onMouseOver="this.style.background='#96F7F4'" onMouseOut="this.style.background=''">
		<td width="100%" align="left">   <img src="../images/folder.gif" width="16" height=
"16" border="0"><?php echo $rows[1] ?>-----------------<a href="edit_depart.php?id=<?php echo $rows[0]; ?>">
修改</a>||<a href="del_depart_chk.php?id=<?php echo $rows[0]; ?>" onClick="return cfm();">删除</a></td></tr>
<?php
		}
	/*当前部门有下属部门时*/
		else{
			$m += 1;
?>
	<!--鼠标划过样式-->
	<tr onMouseOver="this.style.background='#96F7F4'" onMouseOut="this.style.background=''">
❷		<td> <a href="Javascript:ShowMe(img<?php echo $m; ?>,OpenMe<?php echo $m; ?>)">
<img src="../Images/jia.gif" border="0" alt="展开" id="img<?php echo $m; ?>"><?php echo $rows[1];?>
</a>-----------------<a href="edit_depart.php?id=<?php echo $rows[0]; ?>">修改</a>||删除</td></tr>
<?php
❸			list_menu($rows[0],$wid,$m);
		}}
?>
	</table></td></tr></table></td></tr>
<?php}?>
```

代码贴士

❶ onMouseOver…onMouseOut：onMouseOver 表示鼠标指针停留在当前元素上时的状态，onMouseOut 事件表示鼠标指针离开当前元素时的状态。本行用不同的颜色表示两种不同的事件。

❷ 当前部门有下属部门时，单击图片或文字链接，触发一个 js 事件，显示隐藏域。

❸ 调用自身函数。

在这个模块中，同时使用了另一个自定义函数 isbool($id,$dname)，该函数将$dname 参数和 tb_depart 表中的部门名称一一比较，如果重复，则返回 true，否则返回 false。$id 是在修改部门时，先去掉要修改的部门的名称再进行比较。在信息处理页中，根据返回的值进行操作。isbool()函数的程序代码如下：

例程20　代码位置：光盘\TM\03\oa\admin\inc\func.php

```
function isbool($id,$dname){
	/*连接数据库*/
	$conn = mysql_connect("localhost","root","root");
	mysql_select_db("db_office",$conn);
	mysql_query("set names gb2312");
	/*****************/
	$sqlstr = "select * from tb_depart where id != $id d_name = '$dname'";	//创建查询 SQL 语句
	$result = mysql_query($sqlstr,$conn);	//执行 SQL 语句
	if(mysql_num_rows($result)>0)	//判断是否有记录集
		$isbool = true;	//如果有，$isbool = true
	else
		$isbool = false;	//否则 $isbool = false
	return $isbool;	//返回$isbool
}
```

3.10.3　部门查看的实现过程

部门查看模块使用的数据表：tb_depart

通过选择导航栏中的“部门管理”/“查看部门”选项，进入部门查看页面。在部门查看页面中显示部门设置及架构情况，如图 3.29 所示。

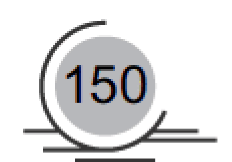


图 3.29　部门查看页面

在部门查看页面中，使用到了递归函数。首先查找根部门（根部门的条件是 top_depart = 0），并返回根部门的结果集；接着使用 while 循环输出所有根部门，每输出一个根部门时，都要查看当前部门是否有子部门（下属部门）。如果没有子部门，那么直接输出部门名称和对部门的操作（修改和删除）；如果根部门有子部门，则调用递归函数（list_menu()）。该页面的程序代码如下：

例程21　代码位置：光盘\TM\03\oa\admin\bmgl\show_depart.php

```
<?php
	session_start();	//开启 session 支持
	include "../inc/chec.php";	//包含权限检查文件
```



```
	include "../conn/conn.php";	//包含数据库链接文件
	include "../inc/func.php";	//包含自定义函数文件
?>
<script src="../js/admin_js.js"></script>	//引入 js 脚本文件
<link href="../css/style.css" rel="stylesheet">	//引入外部 css 样式
<body onLoad="ShowMe(img1,OpenMe1)">	//onLoad 事件，表示当页面被载入时调用 ShowMe()方法
<!--查看部门信息-->
<table width="100%" height="25"  border="0" cellpadding="0" cellspacing="0">
<?php
	/*找到根部门*/
	$sqlstr = "select * from tb_depart where top_depart = 0";
	$result = mysql_query($sqlstr,$conn);
	/*隐藏域 id 号*/
	$m = 1;
	/*循环输出所有根部门*/
	while($rows = mysql_fetch_row($result)){
		$wid = 100;
	/*查看下属部门*/
		$sqlstr1 = "select * from tb_depart where up_depart = ".$rows[0];
		$result1 = mysql_query($sqlstr1,$conn);
		$nu = mysql_num_rows($result1);
	/*当前部门没有下属部门时*/
		if(!$nu){
		?>
	<tr onMouseOver="this.style.background='#96F7F4'" onMouseOut="this.style.background=''">
		<td>    <a href="<?php echo $PHP_SELF; ?>"><img src="../Images/ folder.gif" border="0" alt="展开" id="img<?php echo $m; ?>"><?php echo $rows[1]; ?></a>-----------------<a href="edit_depart.php?id=<?php echo $rows[0]; ?>">修改</a>||<a href="del_depart_chk.php?id=<?php echo $rows[0] ?>" onClick="return cfm();">删除</a></td>
	</tr>
		<?php
	/*当前部门有下属部门时*/
		}else{
?>
	<tr onMouseOver="this.style.background='#96F7F4'" onMouseOut="this.style.background=''">
		<td><a href="Javascript:ShowMe(img<?php echo $m; ?>,OpenMe<?php echo $m; ?>)"><img src= "../Images/jia.gif" border="0" alt="展开" id="img<?php echo $m; ?>"><?php echo $rows[1]; ?></a>-----------------<a href="edit_depart.php?id=<?php echo $rows[0]; ?>">修改</a>||删除</td>
	</tr>
<?php
/*输出同级部门，调用递归函数*/
	list_menu($rows[0],$wid,$m);
	$m += 1;
	}}
/*循环结束*/
?>
```

1．修改部门

在查看部门页面中，可以对部门的名称、上级部门及主要职责等相关信息进行修改。单击要修改部门名称后面的“修改”超链接，进入修改部门页面，如图 3.30 所示。



图 3.30　修改部门页面

修改部门页面（edit_depart.php）中的主要表单元素如表 3.9 所示。

表 3.9　修改部门页面中的表单元素

名　称	元 素 类 型	重 要 属 性	含　义
d_name	Text	size="20" value="<?php echo $rows[d_name] ?>"	部门名称

u_id	select	<?php
		while($tmprows = mysql_fetch_array($tmpresult)){
		if($tmprows[id] == $rows[top_depart]){
		echo "<option value='".$tmprows[id]."'
		selected>".$tmprows[d_name]."</option>";
		}else
		echo "<option value='".$tmprows[id]."'>".$tmprows[d_name]."</option>";
		}?>
			列出所有部门

remark	textarea	<?php echo $rows[remark] ?>	备注信息
id	hidden	value="<?php echo $rows[id] ?>"	修改部门的 id
Submit	Submit	name="Submit" value="修改" onclick="return a_check();"	“修改”按钮

当管理员修改完相应的信息后单击“修改”按钮提交修改信息，系统转到处理页（edit_depart_chk.php），首先判断输入的部门名称是否重复。如果重复，则返回；如果不重复，则继续执行，确定修改后的上级部门和根部门。关键代码如下：

例程22　代码位置：光盘\TM\03\oa\admin\bmgl\edit_depart_chk.php

```
<?
    session_start();                                    //开启 session 支持
    include "../inc/chec.php";                          //包含权限检查文件
    include "../conn/conn.php";                         //包含数据库链接文件
    include "../inc/func.php";                          //包含数据库链接文件
/*判断输入部门名称是否重复*/
if(isbool($_POST[id],$_POST[d_name])){
    echo "<script>alert('名称已存在，请重新输入!!');history.go(-1);</script>";
    exit();
}
/*修改部门，确定上级部门和根部门*/
if($_POST[u_id] != 0){
```



```
/*查找上级部门的根部门*/
        $sqlstr = "select top_depart from tb_depart where id = ".$_POST[u_id];
        $result = mysql_query($sqlstr,$conn);
        $rows = mysql_fetch_array($result);
/*如果存在上级部门，就取得上级部门的根部门*/
        if ($rows[top_depart] != 0)
                $top_depart = $rows[top_depart];
/*如果不存在，将上级部门作为根部门*/
        else
                $top_depart = $_POST[u_id];
}else
/*如果没有上级部门，那么就将自身定为根部门*/
        $top_depart = 0;
/*创建、执行修改部门的 SQL 语句*/
        $sqlstr = "update tb_depart set d_name = '".$_POST[d_name]."',top_depart = ".$top_depart.", up_depart =
".$_POST[u_id].", remark = '".$_POST[remark]."' where id = ".$_POST[id];
        $result = mysql_query($sqlstr,$conn);
        re_message($result,"show_depart.php");
?>
```

2．删除部门

如果要取消某个部门，可以将该部门删除。在删除部门时，只能对最底层的部门进行删除，如果部门下面有子部门，其部门后面的“删除”超链接不可用。删除功能的实现相对比较简单一点，单击“删除”超链接后，系统提示“是否确认删除部门”，如果单击“确认”按钮，系统将转到处理页（del_depart_chk.php）进行删除处理。程序代码如下：

例程23　代码位置：光盘\TM\03\oa\admin\bmgl\del_depart_chk.php

```
<?php
session_start();                                              //开启 session 支持
include "../inc/chec.php";                                    //包含权限查看文件
include "../conn/conn.php";                                   //包含数据库链接文件
include "../inc/func.php";                                    //包含自定义文件
$sqlstr = "delete from tb_depart where id = ".$_GET[id];      //删除 SQL 语句
$result = mysql_query($sqlstr,$conn);                         //执行 SQL 语句
re_message($result,"show_depart.php");                        //返回最后结果
?>
```

3.10.4　部门添加的实现过程

用户单击导航栏中的“部门管理”/“添加部门”选项，进入添加部门页面（add_depart.php），该页面用于设置新添加部门的相关信息，包括部门名称、上级部门和备注，如图 3.31 所示。

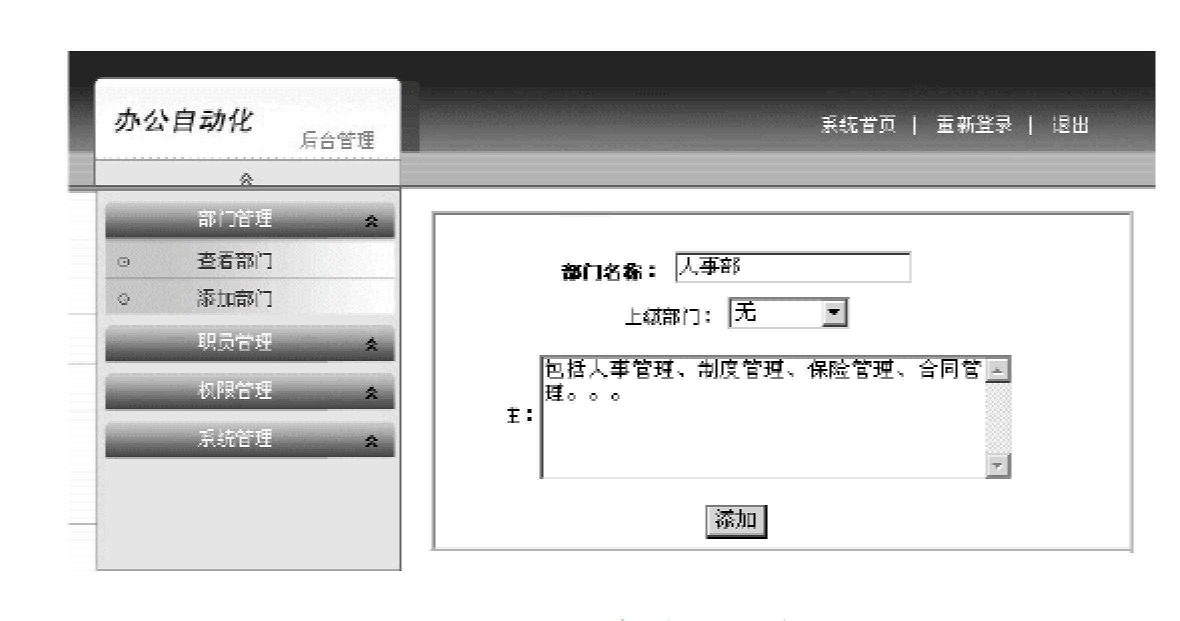


图 3.31　添加部门页面

添加部门页面（add_depart）主要包含一个添加部门表单，表单元素的名称、说明及属性值如

表 3.10 所示。

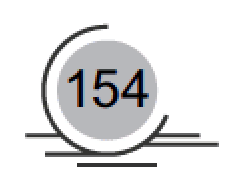
表 3.10　添加部门页面的表单元素

名　称	元 素 类 型	重 要 属 性	含　义
d_name	Text	name="d_name" id="d_name" size="20"	部门名称
u_id	select	<?php while($rows = mysql_fetch_row($result1)) echo "<option value='".$rows[0]."'>".$rows[1]."</option>"; ?>	列出所有部门
remark	textarea	name="remark" cols="40" rows="5" id="remark"	备注信息
Submit	submit	name="Submit" value="添加" onclick="return a_check();"	“添加”按钮

当管理员提交表单后，进入处理页（add_depart_chk.php）进行添加操作。在处理页中，首先判断输入的部门名称是否重复，如果名称符合要求，则进一步确定上级部门和根部门，最后执行 insert 语句进行添加操作。处理页的代码如下：

例程24　代码位置：光盘\TM\03\oa\admin\bmgl\add_depart_chk.php

```
<?
    session_start();                                    //开启 session 支持
    include "../inc/chec.php";                          //包含权限检查文件
    include "../conn/conn.php";                         //包含数据库链接文件
    include "../inc/func.php";                          //包含自定义函数文件
    /*判断输入部门名称是否重复*/
    if(isbool($_POST[d_name])){
        echo "<script>alert('名称已存在，请重新输入!!');history.go(-1);</script>";
        exit();
    }
    /*确定上级部门和根部门*/
    if($_POST[u_id] != "0"){
        $sqlstr = "select top_depart from tb_depart where id = ".$_POST[u_id];
        $result = mysql_query($sqlstr,$conn);
        $rows = mysql_fetch_row($result);
        if ($rows[top_depart] != 0)
            $top_depart = $rows[top_depart];
        else
            $top_depart = $_POST[u_id];
    }
    else
        $top_depart = 0;
    /*添加新部门*/
    $sqlstr = "insert into tb_depart
values('','".$_POST[d_name]."',".$top_depart.",".$_POST[u_id].",'".$_POST[remark]."')";
    $result = mysql_query($sqlstr,$conn);
    /*输出添加结果*/
    re_message($result,"show_depart.php")
?>
```





3.10.5　单元测试

本模块使用了 JavaScript 和自定义函数来显示部门结构列表，在进行测试时发现，当部门被删除到只有一个根部门时，就会弹出错误提示窗口，如图 3.32 所示。

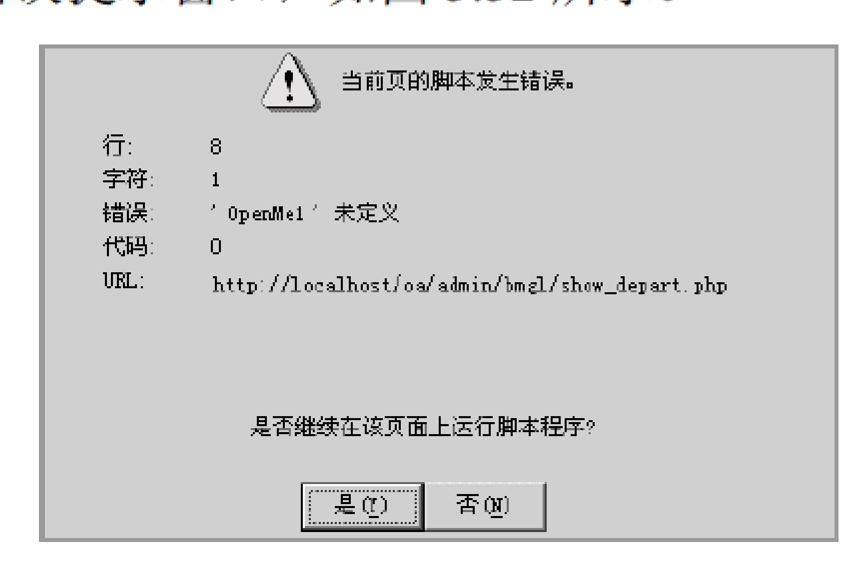


图 3.32　出错信息

单击“是”按钮，其他操作一切正常。经检查发现，在当前页面被载入时调用了 js 脚本，语句如下：

```
<body  onLoad="ShowMe(img1,OpenMe1)">
```

在部门被删到只有一个时，变量 OpenMe1 就不再存在，调用函数 ShowMe()时就会报错，这时删除“onLoad="ShowMe(img1,OpenMe1)"”，错误就不会再出现。

3.11　系统管理模块设计

系统管理模块在后台系统中占据着非常重要的位置，后台其他的模块都是针对前台功能而设计的。而系统管理是针对系统本身的一些信息、情况而设计的，一般包括数据备份、管理员信息修改等，甚至可以对操作系统的配置文件进行操作，可以说是管理员的管理员，是后台系统中必不可少的组成部分。

3.11.1　系统管理模块概述

系统管理模块包括如下功能。

☑　系统日志：记录了前、后台用户登录时的相关信息。

☑　数据备份：可以对整个数据库进行备份、恢复和删除。

☑　修改密码：修改后台管理员登录密码。

系统管理用例图如图 3.33 所示。

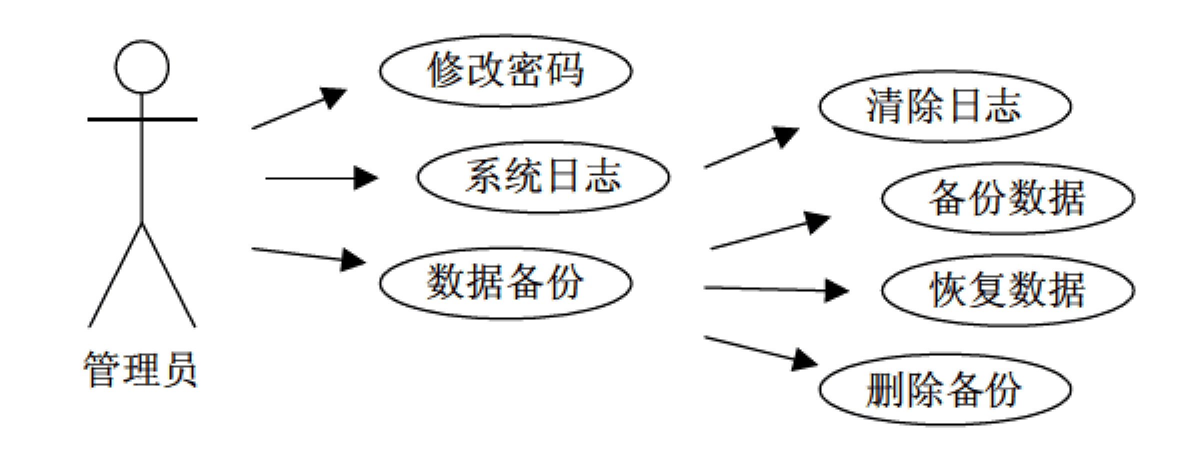
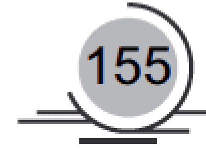


图 3.33　系统管理用例图

3.11.2　系统管理模块技术分析

在系统管理模块中，保存数据的方式使用了数据库和文件，关于数据库技术已经介绍很多了，本



节主要讲解一下文件及文件夹的相关操作。

对文件的操作主要分 3 步：打开文件，读取/写入文件和关闭文件。

1．打开文件

打开文件使用的是 fopen()函数，该函数的一般格式如下：

```
int fopen (string filename,string mode );
```

参数 filename 是要打开的包含路径的文件名，可以是相对路径或绝对路径。如果没有任何前缀，则表示打开的是本地文件；参数 mode 是打开文件的方式，常用的有 a（追加）、w（只写）和 r（只读）。代码如下：

```
$f_open = fopen("../bak/remark.sql","a+");
```

2．读取/写入文件

读取文件的函数有很多，最常用的就是按行读取（fgets()函数）和全部读取（readfile()函数和 fpassthru()函数）。

fgets()函数的一般格式如下：

```
string fgets ( int filename [ ,int length ]) ;
```

参数 filename 是被打开的文件，参数 length 是要读取的数据长度。该函数能够实现从 filename 指定的文件中读取一行并返回长度最大值为 length-1 个字节的字符串。在遇到换行符、EOF 或者读取了 length-1 个字节后停止。如果没有指定 length 的长度，默认值是 1KB。代码如下：

```
<?php
	$f_open = fopen ("../bak/remak.txt","r") ;			//打开文本
		echo fgets ( $f_open) ;				//读取其中的数据
	fclose ( $f_open ) ;
?>
```

readfile()函数的一般格式如下：

```
int readfile ($filename );
```

参数$filename 为要打开的文件名，使用该函数时，无须进行第 1、3 步，也无须输出语句 echo，直接就可以读取文件的全部内容并显示在页面中。

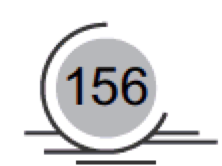
3．关闭文件

完成对文件的操作后，就可以关闭文件了。关闭文件的命令十分简单，格式如下：

```
fclose ( $open);
```

其中参数$open 是要关闭的文件资源。

在本模块中，还用到了两个自定义函数 c_log()和 show_file()。c_log()函数的作用是删除日志文件，程序代码如下：



例程25　代码位置：光盘\TM\03\oa\admin\inc\func.php

```
function c_log(){
	$filename="../log.txt";                              //日志文件路径
	if(file_exists($filename))                           //判断文件是否存在
		unlink($filename);                               //如果文件存在，删除
	else                                                 //如果文件不存在，返回上层操作
		echo "<script>alert('暂无系统日志！');history.go(-1);</script>";
}
```

show_file()函数的作用是返回一个目录下的所有文件名列表，并以数组的形式返回。show_file()函数的程序代码如下：

例程26　代码位置：光盘\TM\03\oa\admin\inc\func.php

```
function show_file(){
	$folder_name = "../bak";                             //数据备份目录
	$d_open = opendir($folder_name);                     //打开备份目录
	$num = 0;                                            //数组下标
	while($file = readdir($d_open)){                     //循环读取文件
		$filename[$num] = $file;                         //将文件名存为数组
		$num++;
	}
	closedir($d_open);                                   //关闭目录
	return $filename;                                    //返回文件数组
}
```

3.11.3　系统日志的实现过程

通过导航栏中的“系统管理”/“系统日志”可进入系统日志页面。在该页面中显示系统日志信息，在系统日志中，记录着普通用户和后台管理员所有人员登录系统时的信息，包括登录时间、用户账号、登录 IP 等。这些信息是在用户登录时系统自动添加到数据表中的（关于用户登录的实现请参见本书附带的光盘）。系统日志页面的运行结果如图 3.34 所示。



图 3.34　系统日志页面的运行结果

显示日志是对文件进行读取，定位到日志文件 log.txt 后，使用 fgets()函数读取日志内容，并显示在页面中。程序代码如下：

例程27　代码位置：光盘\TM\03\oa\admin\xtgl\slog.php

```
<?php
	session_start();                                     //开启 session 支持
	include "../inc/chec.php";                           //包含权限检查文件
?>
<script src="../js/admin_js.js"></script>                //引入 js 脚本文件
<link href="../css/style.css" rel="stylesheet">          //引入外部 css 样式文件
…
	<tr>
		<td colspan="4" height="30" align="center" valign="middle"><a href="del_slog_chk.php" onclick=
"return del_chk();">清除日志</a></td>
```

```
    </tr>
…
<?php
    $filename =   "../log.txt";                              //日志文件路径
    if($f_open = fopen($filename,"r"))                       //是否以只读形式打开文件
    {
        while($str = fgets($f_open,255)){                    //按行读取日志文件内容
            $chr = split(",",$str);                          //以“,”为分隔符，返回数组形式的记录集
            echo "<tr>";
            for($i = 0; $i < count($chr); $i++){             //循环输出字段值
                echo "<td align='center' height='25'>".$chr[$i]."</td>";      }
            echo "</tr>";}
        fclose($f_open);                                     //关闭文件
        } else                                               //如果没有日志文件时
            echo "<script>alert('还没有日志文件！ ');history.go(-1);</script>";
?>
</table>
</td></tr></table>
```

对于无用的日志信息需要定期清除，通过页面中的“清除日志”超链接清空日志，实际上就是删除日志文件。程序代码如下：

例程28　代码位置：光盘\TM\03\oa\admin\xtgl\del_slog_chk.php

```
<?php
    session_start();                                         //开启 session 支持
    include "../inc/chec.php";                               //包含权限检查文件
    include "../inc/func.php";                               //包含自定义函数文件
    c_log();                                                 //调用自定义函数
    echo "<script>alert('删除成功！ ');location='data_stock.php';</script>";
?>
```

3.11.4　数据备份的实现过程

在病毒肆虐的今天，数据备份的重要性是毋需置疑的。没有任何一家网站、一个系统敢说自己是绝对安全的，只要稍不留神就会中招。本着对读者负责的态度，在 3.13 节的技术专题中，将详细地介绍 MySQL 数据的备份与恢复，以及如何在 PHP 中对 MySQL 进行备份操作。这里先简单了解一下本系统数据备份的实现过程。数据备份页面的运行结果如图 3.35 所示。

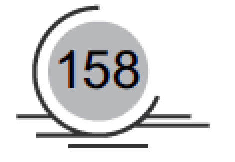


图 3.35　数据备份页面的运行结果

当单击“备份数据”按钮时，系统执行 exec()函数，该函数的作用是执行系统命令，通过系统命令来备份数据。数据备份的程序代码如下：

例程29　代码位置：光盘\TM\03\oa\admin\xtgl\data_stock_chk.php

```
<?php
    session_start();                                //开启 session 支持
```



```
	include "../inc/chec.php";                                //包含权限检查文件
	$mysqlstr = "D:\\\AppServ\\MySQL\\bin\\mysqldump -uroot -h192.168.1.149 -proot --opt -B db_office
> ../bak/".$_POST[b_name];                                    //数据库备份语句
	exec($mysqlstr);                                          //执行系统命令
	echo "<script>alert('备份成功');location='data_stock.php'</script>";
?>
```

恢复数据和备份数据的执行过程大同小异，都需要使用 exec()函数执行系统命令，只是命令执行的操作是完全相反的。恢复数据的程序代码如下：

例程30　代码位置：光盘\TM\03\oa\admin\xtgl\rebak_stock_chk.php

```
<?php
	session_start();                                          //开启 session 支持
	include "../inc/chec.php";                                //包含权限检查文件
	$mysqlstr = "D:\\\AppServ\\MySQL\\bin\\mysql -uroot -h192.168.1.149 -proot db_office < ../bak/".$_POST
[r_name];                                                     //数据恢复命令
	exec($mysqlstr);                                          //执行系统命令
	echo "<script>alert('恢复成功');location='data_stock.php'</script>";
?>
```

删除数据备份实质就是删除备份文件，先使用自定义函数 show_file()返回一个文件名列表数组，再使用 for 循环语句，逐一删除备份文件。程序代码如下：

例程31　代码位置：光盘\TM\03\oa\admin\xtgl\del_stock_chk.php

```
<?php
	session_start();                                          //开启 session 支持
	include "../inc/chec.php";                                //包含权限检查文件
	include "../inc/func.php";                                //包含自定义函数文件
	$filename = show_file();                                  //调用自定义函数 show_file()，返回文件名列表
	for($num = 2;$num < count($filename);$num++){             //循环删除文件
		unlink("../bak/".$filename[$num]);
	}
	echo "<script>alert('删除成功！');location='data_stock.php'</script>";
?>
```

3.12　开发技巧与难点分析

3.12.1　使用 JavaScript 关联多选列表框

在本系统中，多个模块都使用了这种技术，即在两个多选列表框之间产生关联，从列表框 A 中删除的数据被相应地添加到列表框 B 中，反之从列表框 B 中删除的数据会被添加到列表框 A 中。

在权限管理模块中的用户组设置，就是用这种技术实现的“添加新用户组”功能，首先在左侧的多选列表框中选取要添加的成员，单击“添加组员”超链接，被选中的成员就被添加到右侧列表框，同时删除在左侧列表框中的成员，运行结果如图 3.36 所示。

图 3.36　使用 JavaScript 关联多选列表框

下面来看一下具体的操作，首先创建一个表单，其中两个多选列表框为必选的表单元素，分别命名为 left 和 right，放在左、右两边，其中 left 列表框中的数据是从数据库中读取的，right 列表框则初始为空。程序代码如下：

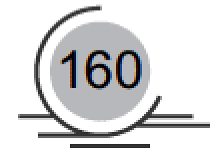
例程32　代码位置：光盘\TM\03\oa\admin\qxgl\add_group.php

```
…
<!--输出左侧列表框-->
<SELECT name="left" size="10" multiple style="width:100px; ">
     <?php
          while($rows = mysql_fetch_row($result)){
               echo "<option value='".$rows[1]."'>".$rows[1]."</option>";       //从数据库中读取列表值
          }
     ?>
</SELECT>
<!-- ---------------------- -->
</td>
<!--"添加"、"删除"超链接-->
<td width="96" align="center" valign="middle">
   <a href="#" onClick="activeList(document.form1.left,document.form1.right)">添加组员&gt;&gt;</a><br>
   <br>
    <a href="#" onClick="activeList(document.form1.right,document.form1.left)">&lt;&lt;删除组员</a></td>
<!--右侧列表框-->
    <td colspan="2" align="center" valign="middle"><select name="right" size="10" multiple style="width:100px; ">
   </select></td>
  </tr>
  <tr>
    <td height="30" colspan="4" align="center" valign="middle">
       <input type="hidden" name="g_list" />
     <!--"添加"按钮-->
       <input type="submit" value="添加" onclick="return glist()" /><input type="reset" value="重置" />
…
```

上面的代码主要是 HTML 和 PHP 语言，和普通的页面没有什么区别，核心的语句就是代码中的加粗部分（不是注释），当单击“添加组员”或“删除组员”超链接时，将触发 JavaScript 脚本事件，执行 activeList()函数。函数代码如下：

例程33　代码位置：光盘\TM\03\oa\admin\js\admin_js.js

```
/*关联多选列表
*headStream：源多选列表
*endStream：目的多选列表
```



```
*/
function activeList(headStream,endStream){
     var valueList = new Array();                              //数组，用来存储要移动数据的 value 值
     var textList = new Array();                               //数组，用来存储要移动数据的 text 值
     var valueTmpText = new Array();                           //数组，用来存储要移动数据的 option 项
     var index = 0;                                            //数组下标
/*存储源列表中被选中的数据*/
     for(var i=0; i<headStream.options.length; i++){
          if(headStream.options[i].selected){                  //判断元素是否被选中
               valueList[index] = headStream.options[i].value;  //存储 value 值
               textList[index] = headStream.options[i].text;    //存储 text 值
               valueTmpText[valueList[index]] = headStream.options[i]; //存储 option 项
               index ++;
          }}
/*向目的列表中添加数据，同时删除对应的源列表数据*/
     for(var i=0; i<textList.length; i++) {
          var foption = document.createElement("option");      //建立新的 option 项
          foption.text = textList[i];
          foption.value = valueList[i];
          endStream.add(foption);                               //向目标列表添加新建 option 项
          headStream.removeChild(valueTmpText[valueList[i]]);   //从源列表中移除 option 项
     }}
```

3.12.2　用户组设置



图 3.37　用户组权限对比

在后台的权限管理模块中，放弃了常用的“读”、“写”这样的常规操作，而是采用了用户组的模式。用户组的设置主要是两方面：一方面是用户组成员，同一用户可以分配在不同的用户组；另一方面是功能列表，可以设置用户组的访问权限，不同的用户组，所看到的功能列表也会不同。这样，可以使用户的权利更加细化，防止居心叵测的用户恶意破坏。用户组 A 和用户组 B 的对比结果如图 3.37 所示。

在功能列表的设置中，首先默认所有的功能都是可访问的，当有的功能需要进行限制时，就将其从用户组中删除。程序代码如下：

例程34　代码位置：光盘\TM\03\oa\admin\qxgl\pur_assign.php

```
<?php
      session_start();                                          //开启 session 支持
      include "../inc/chec.php";                                //包含权限检查文件
      include "../conn/conn.php";                               //包含数据库链接文件
      $sqlstr = "select id,f_name,o_group from tb_list";        //生成 SQL 语句
?>
<script src="../js/admin_js.js"></script>                       //引入 js 脚本文件
<link href="../css/style.css" rel="stylesheet">                 //引入外部 css 样式文件
…
      <tr><td width="75" height="20" align="center" valign="middle">功能</td>
           <td width="75" height="20" align="center" valign="middle">开放组</td>
           <td width="75" height="20" align="center" valign="middle">操作</td></tr>
```

```
<?php
	/*输出功能列表项*/
	$result = mysql_query($sqlstr,$conn);
	while($rows = mysql_fetch_row($result)){
		echo "<tr>";
		for($i = 1;$i < count($rows); $i++){
			echo "<td align=center valign=middle>".$rows[$i]."</td>";
		}
		echo "<td align=center valign=middle><a href='modify_assign.php?id=".$rows[0]."'>修改</a></td>";
		echo "</tr>";
	}
/**********************************/
?>
</table>
</td></tr></table>
```

本节是 3.12.1 节关联多选列表框的延伸，是具体应用关联列表实现的一项功能。在实际的应用中，应多开动脑筋，才能用平凡的技术实现不平凡的功能。

3.13　MySQL 数据备份专题

关于备份数据的重要性，这里就不再多做介绍。本节先来了解常用数据库备份的方式，再学习 MySQL 数据备份与恢复的方法，最后看一看在 PHP 中是如何对 MySQL 数据进行备份的。

1．数据库备份类型

（1）完全备份

这是大多数人常用的方式，它可以备份整个数据库，但需要花费更多的时间和空间，所以一般推荐一周做一次。

（2）事务日志备份

事务日志就是对数据库操作的记录，它记录数据库的改变，备份时只需要备份自上次备份以来对数据库所做的改变，所以只需要很少的时间。推荐每小时一次。

（3）差异备份

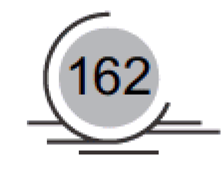
差异备份也叫增量备份。它是另一种只备份一小部分数据库的方法，但它不使用事务日志，相反，它使用整个数据库的一种新映象。差异备份比最初的完全备份小，因为它只包含自上次完全备份以来所改变的数据库。其优点是存储和恢复速度快。推荐每天做一次。

（4）文件备份

数据库可以由硬盘上的许多文件构成。如果这个数据库非常大，恐怕一个晚上也不能将它备份完，可以每天备份数据库的一部分。由于一般情况下数据库不会大到必须使用多个文件存储，所以这种备份不是很常用。

2．MySQL 数据备份与恢复

MySQL 中的备份是非常灵活的，既可以对单独的数据表进行备份，也可以对整个数据库进行备份，下面分别进行介绍。





（1）数据表备份

备份数据表的语法如下：

```
SELECT * INTO OUTFILE 'file_name' FROM tbl_name
```

☑ file_name：备份的文件名，如 user.sql。

☑ tb1_name：备份的数据表名，如 tb_users。

使用该备份语句首先要登录到 MySQL，并进入到相对应的数据库中，如要备份的表 tb_users 是属于数据库 db_office 的，备份文件名为 users.sql，则完整的备份流程如图 3.38 所示。

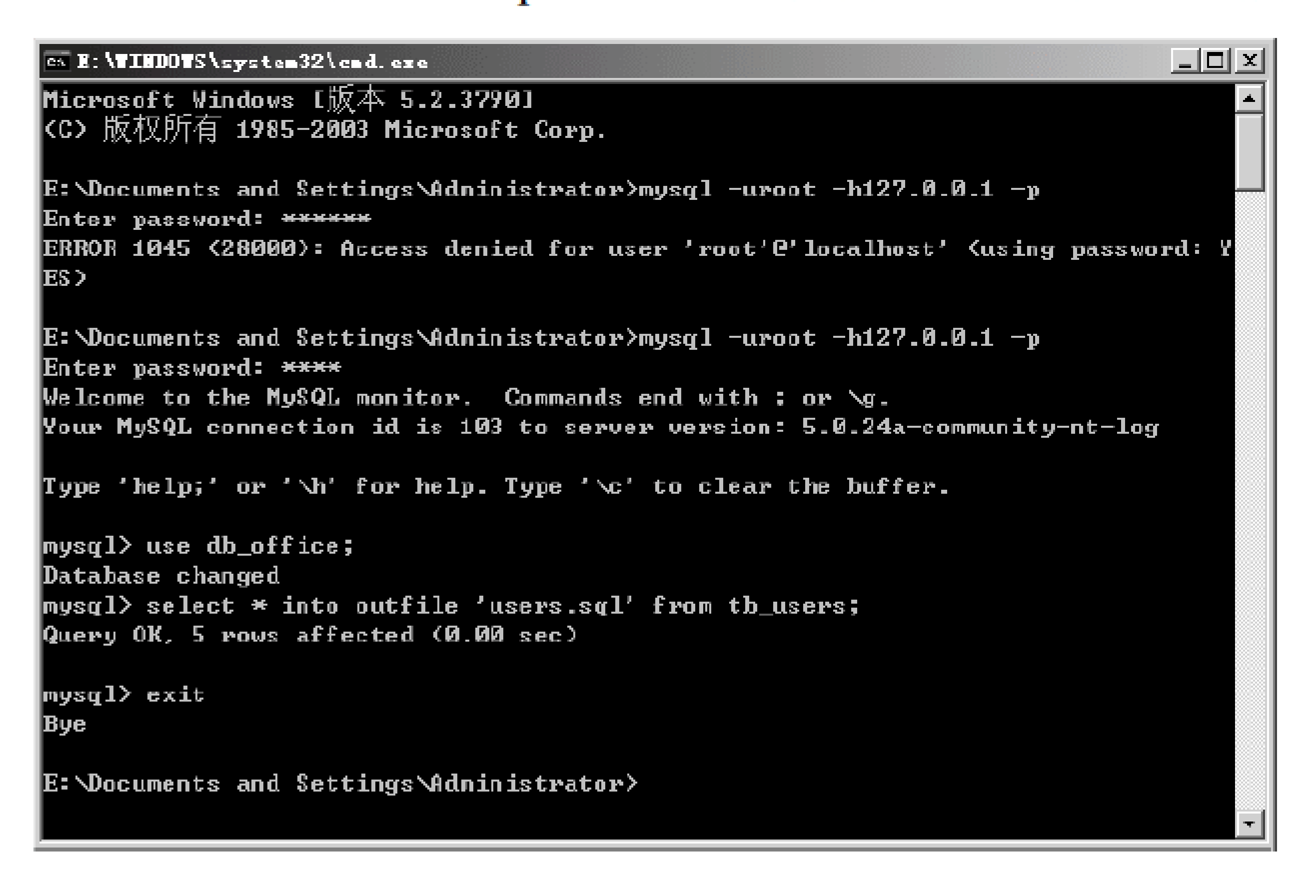


图 3.38　数据表备份

备份后的文件保存在数据表所在目录中，如本例中表 tb_users 所在的目录为 D:\AppServ\MySQL\data\db_office，则备份的 sql 文件就保存在这个路径下。

如果要恢复备份文件，代码如下：

```
LOAD DATA INFILE 'user.sql' REPLACE into table 'tb_users'
```

（2）数据库备份

数据库备份时，可以不用登录到 SQL 中，直接使用 mysqldump 命令即可。该命令格式如下：

```
mysqldump -uroot -proot db_office > c:\\remark.txt
```

备份的数据表，实际就是一个完整的建库流程，如果要恢复备份，直接用 MySQL 命令即可。语句如下：

```
mysql -uroot -proot db_office < c:\\remark.txt
```

mysqldump 语句的功能十分强大，备份只是其中的一个功能，感兴趣的读者可以参考 mysqldump 的帮助文档。

3．在 PHP 中备份数据库

在 PHP 中备份数据库，主要通过 exec()函数执行系统命令。

exec()函数的格式如下：

```
string exec ( string command )
```

参数 command 为要执行的系统命令。这里为备份数据表的命令，代码如下：

```
exec("D:\\\AppServ\\MySQL\\bin\\mysqldump -uroot -h192.168.1.149 -proot --opt -B db_office > ../bak/remark.sql");
```

注意 虽然在安装时 MySQL 已经设置了环境变量，但如果要在 PHP 中执行 mysqldump、mysql 命令，则必须给出命令的完整路径；否则，就会出现文件大小为 0KB 的结果。

3.14 本 章 总 结

本章通过一个完整的办公自动化管理系统开发过程，向读者讲述了系统开发的几个重要环节，还包括几个主要模块的功能实现，最后又着重介绍了 MySQL 备份的相关知识。通过本章的学习，读者对如何开发一个真实的项目会有进一步的了解。相信读者通过自己动手实践，完成一个功能相似的办公自动化管理系统是完全不成问题的。

第 4 章

网络在线考试系统

（Apache+PHP+Ajax+SQL Server 2000 实现）

随着科技的发展，网络技术已经深入到人们的日常生活中，同时带来了教育方式的一次变革。而网络考试则是一个很重要的方向。基于 Web 技术的网络考试系统可以借助于遍布全球的 Internet 进行。因此考试既可以在本地进行，也可以在异地进行，大大拓展了考试的灵活性，并且缩短了传统考试要求老师打印试卷、安排考试、监考、收集试卷、评改试卷、讲评试卷和分析试卷这个漫长而复杂的过程，使考试更趋于客观、公正。本章介绍了一个具有在线考试、即时阅卷、成绩查询以及考题和考生信息管理等功能的网络在线考试系统。通过阅读本章，可以学习到：

- ⏭ 掌握网络在线考试的开发过程
- ⏭ 掌握应用 SQL Server 2000 创建数据库、数据表的方法
- ⏭ 掌握连接 SQL Server 数据库的不同方式
- ⏭ 实时显示时间的两种不同方法
- ⏭ 掌握 Ajax 无刷新技术

4.1 开发背景

随着教育改革的不断深入，近几年来出现了一些新的教育形式，如网上授课、网上考试等。×××高校是一所师资力量非常雄厚的学校，随着院校的扩招，学生数量不断增加。为了适应新形式的发展，改变传统的教学模式，方便学生随时随地地对自己的学习情况进行检测，减轻教师的工作压力，现委托其他公司开发网上在线考试系统。这种无纸的网络考试系统，使考务管理突破时空限制，提高考试工作效率和标准化水平，使学校管理者、教师和学生可以在任何时候、任何地点通过网络进行考试。网络在线考试系统已经成为教育技术发展与研究的方向。

4.2 系统分析

4.2.1 需求分析

随着计算机技术的发展和网络技术的日益成熟，通过网络进行信息交流已成为一种快捷的交互方式。在这种网络环境下，学校或考试机构希望通过建立网络在线考试网站来扩大知名度、降低管理成本和减少人力、物力的投资，从而为考生提供更全面、更灵活的服务，并全面、准确地对考试进行跟踪和评价。与此同时，考生希望根据自己的学习情况进行测试，并能够得到客观、科学的评价；教务人员希望能够有效地改进现有的考试模式，提高考试效率。

通过实际情况的调查，要求网络在线考试系统具有以下功能：

- ☑ 界面设计美观大方、方便、快捷、操作灵活，树立企业形象。
- ☑ 要求实现在线考试功能，自动核算考试成绩。
- ☑ 要求提供考试时间倒计时功能，使考生实时了解考试剩余时间。
- ☑ 要求系统自动阅卷，保证考试成绩真实有效。
- ☑ 要求考生凭准考证号查询考试成绩，以保证信息安全。
- ☑ 系统运行稳定、安全可靠。
- ☑ 要求对考生及考题信息进行严格管理。

4.2.2 可行性分析

可行性分析的目的就是要用最小的代价在尽可能短的时间内确定问题是否能够解决。通过分析解法的利弊，来判定系统目标和规模是否现实，系统完成后所能带来的效益是否达到值得去投资开发这个系统的程度。网络在线考试系统的可行性可从以下两方面考虑。

1. 经济可行性

定期地组织考试是各个院校及时掌握学生学习成绩的有效方式。利用网络在线考试系统，一方面可以节省人力资源，降低考试成本；另一方面，在线考试系统能够快速进行考试和评分，体现出考试的客观与公正性。



2．技术可行性

开发一个网络在线考试系统，涉及的最核心的技术问题就是如何实现在不刷新页面的情况下实时显示考试时间及剩余时间，并做到到达考试结束时间时自动提交试卷的功能。通过 Ajax 技术可以轻松实现这些功能，这为网络在线考试系统的开发提供了技术保障。

4.3　系 统 设 计

4.3.1　系统目标

根据前面所作的需求分析及用户的需求可知，网络在线考试系统属于中小型软件，在系统实施后应达到以下目标：

- ☑ 采用开放、动态的系统架构，加强用户与网站的动态交互性。
- ☑ 具有空间性。被授权的用户可以在异地登录网络在线考试系统，无须到指定地点进行考试。
- ☑ 操作简单方便、界面简洁美观。
- ☑ 系统提供考试时间倒计时功能，使考生实时了解考试剩余时间。
- ☑ 随机抽取试题。
- ☑ 实现自动提交试卷的功能。当考试时间到达规定时间时，如果考生还未提交试卷，系统将自动交卷，以保证考试严肃、公正地进行。
- ☑ 系统自动阅卷，保证成绩真实准确。
- ☑ 考生可以查询考试成绩。
- ☑ 对考生注册信息进行管理。
- ☑ 系统运行稳定、安全可靠。

4.3.2　系统功能结构

网络在线考试系统的前台功能结构如图 4.1 所示。

网络在线考试系统的后台功能结构如图 4.2 所示。

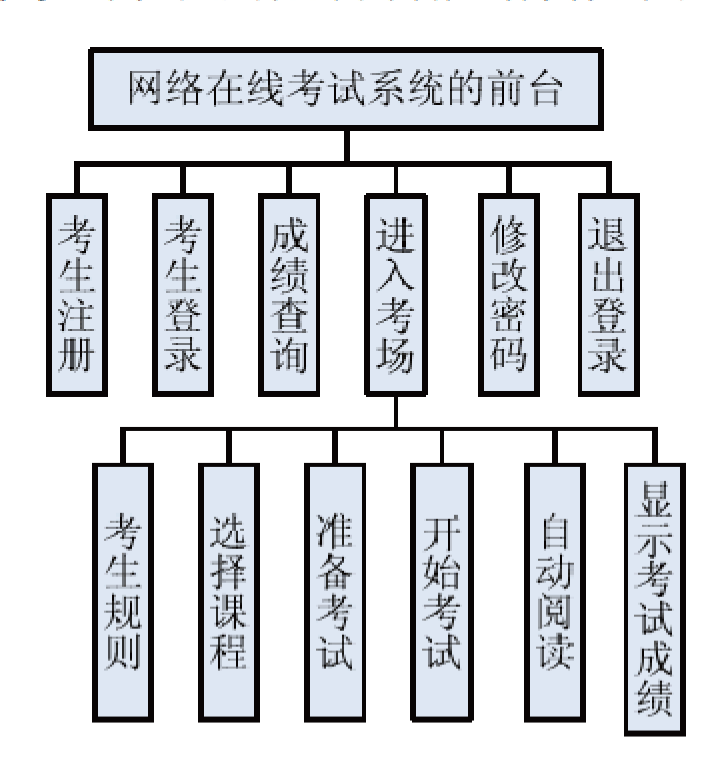


图 4.1　网络在线考试系统的前台功能结构图

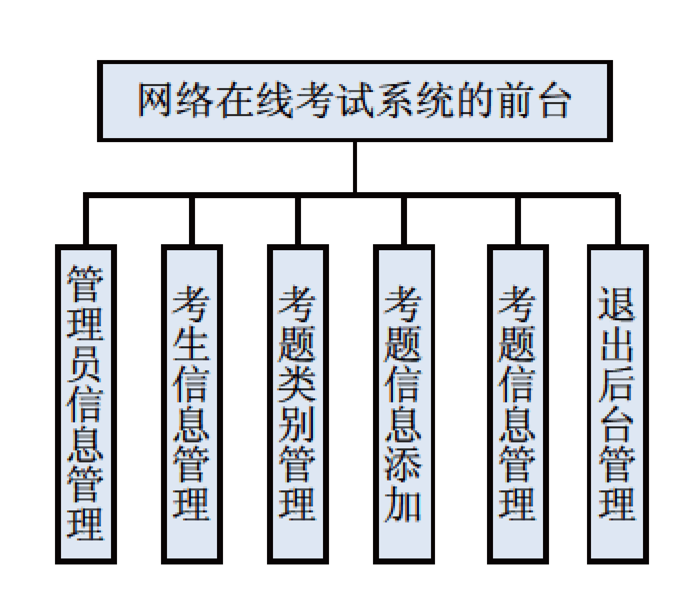
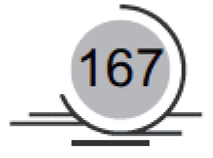


图 4.2　网络在线考试系统的后台功能结构图



4.3.3　系统流程图

为了读者能够更好地学习，下面给出网络在线考试网的网站流程概述及系统流程图。

考生在网络在线考试系统中通过注册为网站用户，登录网站进行相关操作。考生登录后，可以进行在线考试、查询成绩和修改个人密码的操作。在考试前，考生需要阅读考试规则、选择考试套题后开始考试；考试时间结束时，考生提交试卷；提交试卷的同时，系统将自动返回本次考试的考试结果。网站的管理员通过登录模块可以登录到网站的后台系统，对考生信息、考试信息、管理员信息进行管理。网络在线考试系统的系统流程如图 4.3 所示。

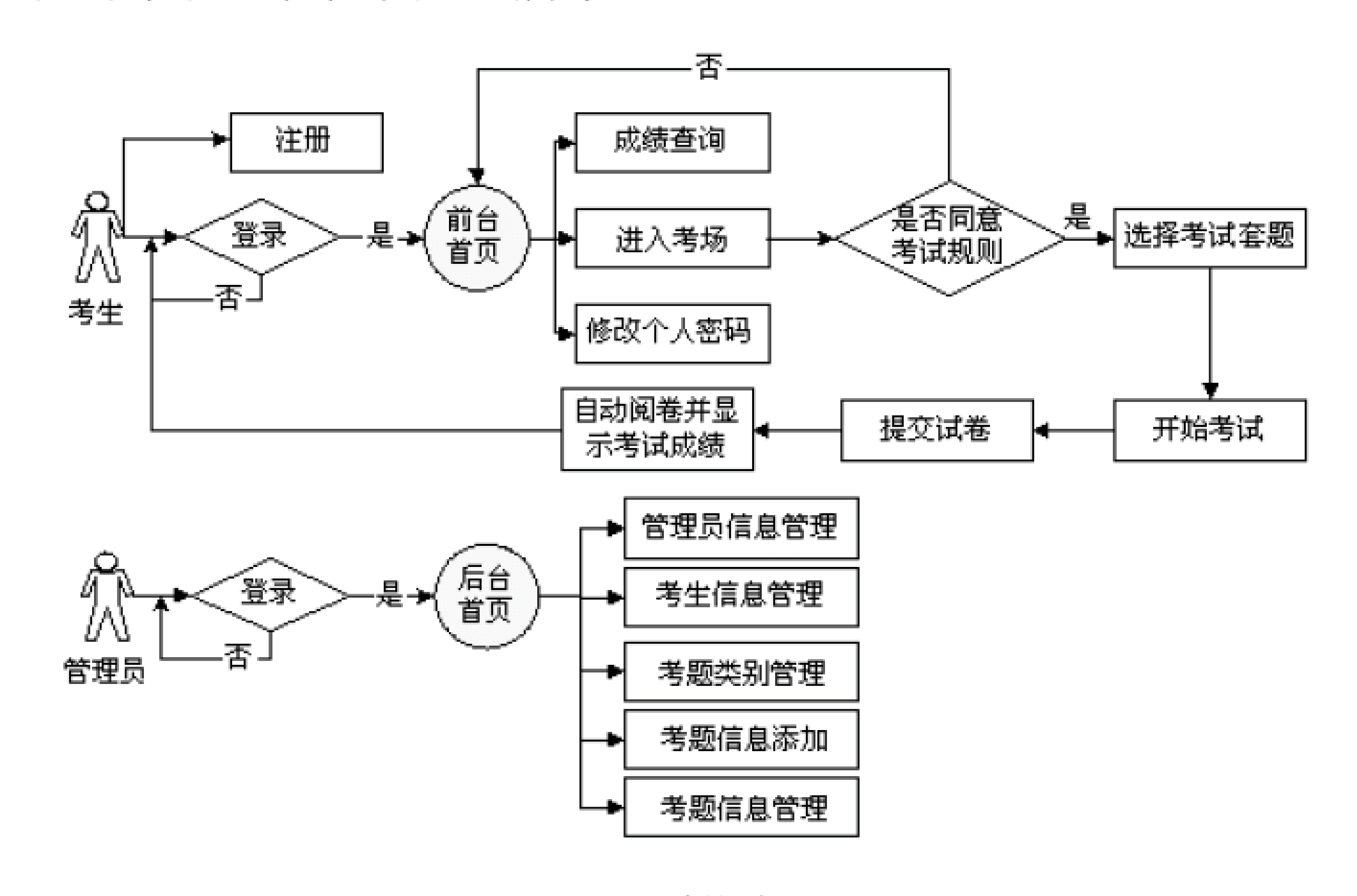


图 4.3　系统流程图

4.3.4　系统预览

网络在线考试系统由多个页面组成，下面仅列出几个典型页面，其他页面参见光盘中的源程序。

考生登录页面如图 4.4 所示，该页面主要用于考生登录，实现在线考试及考试成绩查询等功能。在线考试页面如图 4.5 所示，该页面用于实现在线答题功能，同时提供了显示考试时间和剩余时间及自动提交试卷功能。



图 4.4　考生登录（光盘\TM\04\online_ks\denglu.php



图 4.5　在线考试（光盘\TM\04\online_ks\ksks.php）



考题类别管理页面如图 4.6 所示，该页面主要用于实现显示考题类别的基本信息、添加考题类别和删除考题类别等功能。考题信息管理页面如图 4.7 所示，该页面主要用于管理考试题目信息。



图 4.6　考题类别管理（光盘\TM\04\…\admin\ktlb_gl.php）



图 4.7　考题信息管理（光盘\TM\04\…\admin\ktxx_gl.php）

说明　由于路径太长，因此省略了部分路径，省略的路径是“online_ks\”。

4.3.5　开发环境

在开发网络在线考试系统时，该项目使用的软件开发环境如下。

1．服务器端

- ☑ 操作系统：Windows Server 2003/Linux（推荐）。
- ☑ 服务器：Apache 2.2.8。
- ☑ PHP 软件：PHP 5.2.6。
- ☑ 数据库：SQL Server 2000。
- ☑ 开发工具：Dreamweaver 8。
- ☑ 浏览器：IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

2．客户端

- ☑ 浏览器：IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

4.3.6　文件夹组织结构

在编写代码之前，可以把系统中可能用到的文件夹先创建出来（例如，创建一个名为 images 的文件夹，用于保存网站中所使用的图片），这样不但可以方便以后的开发工作，也可以规范网站的整体架构。笔者在网络在线考试系统中设计了如图 4.8 所示的文件夹组织结构图。在开发时，只需要将所创建的文件保存在相应的文件夹中即可。

online_ks
admin —— 用于存储网站后台文件
conn —— 用于存储数据库连接文件
images —— 用于存储网站图片资源
js —— 用于存储网站使用的自定义函数

图 4.8　网络在线考试系统的文件夹组织结构

4.4　数据库设计

4.4.1　数据库分析

由于网络在线考试系统对于数据的安全性及完整性要求比较高，并且为了增加程序的适用范围，还要保证系统可以拥有存储足够多数据的能力。SQL Server 2000 是一种高性能的关系型数据库管理系统，不仅安全可靠，而且易于操作，已成为在线事务进程和数据仓库等最好的数据库平台之一。综上所述，本系统采用 SQL Server 2000 数据库。

4.4.2　数据库概念设计

根据以上各节对系统所做的需求分析、系统设计，规划出本系统使用的数据库实体分别为考生信息实体、管理员实体、考题类别实体和考题信息实体。下面分别介绍这几个实体的 E-R 图。

1．考生信息实体

考生信息实体包括编号、考生姓名、联系方式、准考证号、考试成绩、考题类别、考试时间、考试状态和考生密码等属性。考生信息实体的 E-R 图如图 4.9 所示。

2．管理员实体

管理员实体包括编号、管理员名称和密码等属性。管理员实体的 E-R 图如图 4.10 所示。

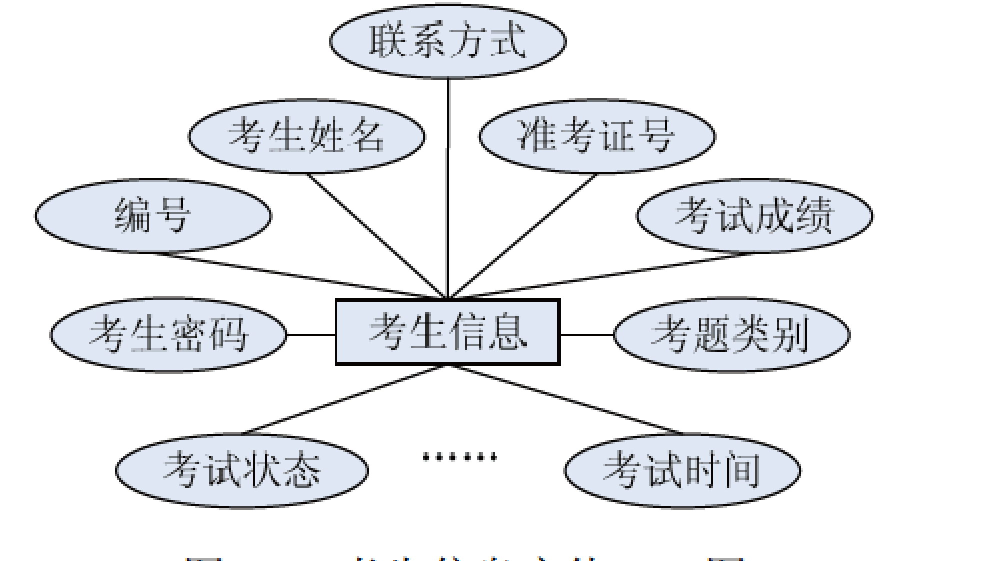


图 4.9　考生信息实体 E-R 图

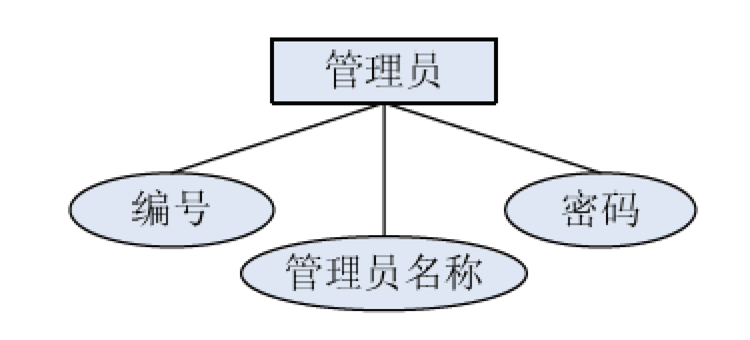


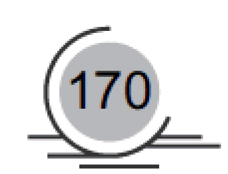
图 4.10　管理员实体 E-R 图

3．考题类别实体

考题类别实体包括编号和类别名称等属性。考题类别实体的 E-R 图如图 4.11 所示。

4．考题信息实体

考题信息实体包括编号、考题类别、考试成绩、考题内容、考题答案、正确答案及所属套题等属



性。考题信息实体的 E-R 图如图 4.12 所示。

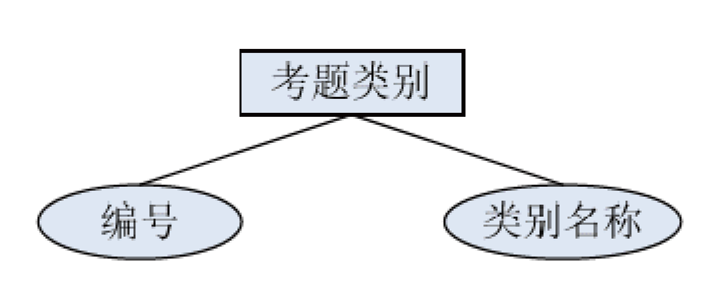


图 4.11　考题类别实体 E-R 图

图 4.12　考题信息实体 E-R 图

4.4.3　数据库物理结构

根据 4.4.2 节的数据库概念设计，可以创建与实体对应的数据表。为了使读者对本系统数据库的结构有一个更清晰的认识，下面给出数据库中所包含数据表的结构图，如图 4.13 所示。

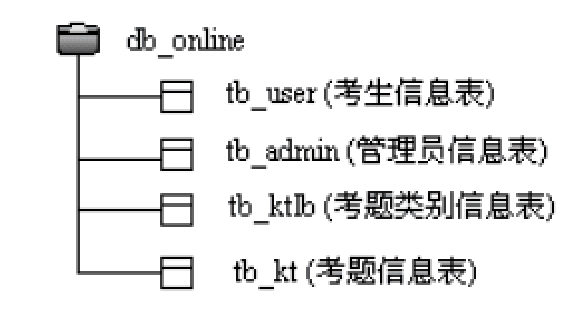


图 4.13　db_online 数据库所包含数据表的结构图

本系统共包含 4 张数据表，下面进行详细介绍。

1．tb_user（考生信息表）

考生信息表用来保存考生信息。该数据表结构如图 4.14 所示。

2．tb_admin（管理员信息表）

管理员信息表用来保存管理员的用户名和密码。该数据表结构如图 4.15 所示。

tb_user

列名	数据类型	长度	允许空	标识	标识种子	描述
online_id	int	4		✓	1	自动编号
online_user	varchar	50				考生姓名
online_tel	varchar	50				考生联系方式
online_number	varchar	50				考生准考证号
online_grade	int	4	✓			考试成绩
online_subject	varchar	50	✓			考题类别
online_date	datetime	8	✓			考试时间
online_address	varchar	100				考生联系地址
online_pt	int	4	✓			考试状态
online_pass	varchar	50				考生密码

图 4.14　考生信息表结构



tb_admin

列名	数据类型	长度	允许空	标识	标识种子	描述
id	int	4		✓	1	自动编号
name	varchar	20	✓			管理员名称
pwd	varchar	20	✓			管理员密码

图 4.15　管理员信息表结构

3．tb_ktlb（考题类别信息表）

考题类别信息表用来保存考题类别。该数据表结构如图 4.16 所示。

4．tb_kt（考题信息表）

考题信息表用来保存考试题目信息和考题答案等相关信息。该数据表结构如图 4.17 所示。

tb_ktlb

列名	数据类型	长度	允许空	标识	标识种子	描述
ktlb_id	int	4		✓	1	自动编号
online_ktlb	varchar	50				考题类别

图 4.16　考题类别信息表结构

tb_kt

列名	数据类型	长度	允许空	标识	标识种子	描述
kt_id	int	4		✓	1	自动编号
kt_lb	varchar	50				考题类别
kt_lx	int	4				考题类型
kt_fs	int	4				考试成绩
kt_nr	varchar	2000				考题内容
kt_daan	varchar	2000	✓			考题答案
kt_zqdaan	varchar	2000				考题正确答案
kt_small_lb	varchar	50				考题所属套题

图 4.17　考题信息表结构

4.5 前台首页设计

4.5.1 前台首页概述

考生通过考生登录模块的验证后，可以登录到网络在线考试系统的前台首页。前台首页主要用于实现前台功能导航，该页面主要包括“考生注册”、“考生登录”、“修改密码”、“成绩查询”、“进入考场”和“退出信息”6 个导航链接。

本案例中提供的前台首页如图 4.18 所示。该页面在本书光盘中的路径为\TM\04\online_ks\index.php。



图 4.18 网络在线考试系统前台首页

4.5.2 前台首页技术分析

PHP 连接 SQL Server 数据库的方法有两种：一是通过 ADO 方式连接；二是通过 mssql_connect()函数连接。下面分别介绍这两种连接方式。

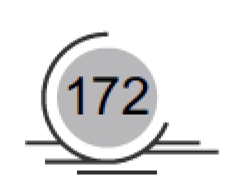
1．通过 ADO 方式连接 SQL Server 数据库

利用 ADO 方式建立与 SQL Server 数据库的连接，与使用 ADO 方式建立与 Access 数据库连接的方法类似，建立连接是在本实例的 conn.php 文件中实现的。代码如下：

```
<?php
$conn=new com("adodb.connection");
$connstr="provider=sqloledb;data source=localhost;uid=sa;pwd=;database=db_online";
$conn->open($connstr);
?>
```

2．通过 mssql_connect()函数连接 SQL Server 数据库

通过 mssql_connect()函数建立与 SQL Server 数据库的连接首先应该配置 php.ini 文件，删除“extension=php_mssql.dll”前面的注释，然后重新启动 Apache 服务器。通过 mssql_connect()函数建立与数据库连接的代码如下：





```
<?php
❶    $conn=mssql_connect("localhost","sa","");                    //建立与 SQL Server 数据库的连接
❷    mssql_select_db("db_online ",$conn);                         //选择数据库
?>
```

代码贴士

❶ mssql_connect()函数：打开一个到 SQL Server 服务器的连接。如果成功，则返回一个 SQL Server 连接标识；失败则返回 false。语法如下：

```
resource mssql_connect ( [string servername [, string username [, string password [, bool new_link]]]] )
```

参数 servername 指 SQL Server 服务器；参数 usernamer 指用户名；参数 password 指用户密码；参数 new_link 指连接标识。

❷ mssql_select_db()函数：选择 SQL Server 数据库。如果成功，则返回 true；失败则返回 false。语法如下：

```
bool mssql_select_db( string database_name [,resource link_identifier])
```

参数 database_name 指数据库名称；参数 link_identifier 指连接标识。

4.5.3 前台首页的实现过程

网络在线考试系统前台首页主要实现了考生登录功能，考生通过准考证号和密码进行登录。

考生登录页面所涉及的重要表单元素如表 4.1 所示。

表 4.1 考生登录页面所涉及的重要表单元素

名 称	元素类型	重要属性	含 义
form1	form	method="post" action="denglu_ok.php"	表单
online_number	text	id="online_number" size="20"	准考证号码
online_pass	password	id="online_pass" size="20"	考生密码
Submit	submit	value="提交"	“提交”按钮

考生在考生登录页面输入准确的准考证号和密码后，单击“提交”按钮，提交表单信息到数据处理页，应用 mssql_query()函数检索考生是否成功登录。如果查询结果为真，则将准考证号存储在 session 变量中；否则弹出提示信息。代码如下：

例程01 代码位置：光盘\TM\04\online_ks\denglu_ok.php

```
<?php
include("conn/conn.php");                                   //连接数据库文件
$online_number=$_POST[online_number];                       //获取准考证号
$online_pass=$_POST[online_pass];                           //获取考生密码
/* ****************************检索考生的准考证号和密码是否正确****************************** */
❶       $query=mssql_query("select * from tb_user where online_number='$online_number' and 
online_pass='$online_pass'");
❷    if(mssql_num_rows($query)>0){                          //如果查询结果为真
❸    session_register(online_number);                       //将准考证号存储在 session 变量中
     echo "<script> alert('登录成功!'); window.location.href='index.php?online=考试规则';</script>";
}else{                                                          //如果没有检索到数据
```

```
        echo "登录失败!";                                          //弹出提示信息
}
?>
```

代码贴士

❶ mssql_query：向与指定的 SQL Server 服务器中的当前活动数据库发送一条 SQL 查询。语法如下：

```
resource mssql_query ( string query [, resource link_identifier] )
```

参数 query 传入的是 SQL 的指令；参数 link_identifier 传入的是由 mssql_connect()函数或 mssql_pconnect()函数返回的连接号，如果省略该参数，则会使用最后一个打开的 MySQL 数据库连接。

❷ mssql_num_rows：获取结果集中行的数目。值得注意的是，该函数仅对 SELECT 语句有效。

❸ session_register：在整个域范围内增加一个 session 变量。语法如下：

```
boolean session_register(string name)
```

参数 name 用于指定新 session 变量的名称。

4.6　考生信息模块设计

4.6.1　考生信息模块概述

考生信息模块主要包括考生注册、考生登录、修改密码 3 个功能。考生首先要注册为网站用户，然后才被授权登录网站进行一系列操作的权限；登录后考生还可以修改个人的密码。考生信息模块的系统流程如图 4.19 所示。

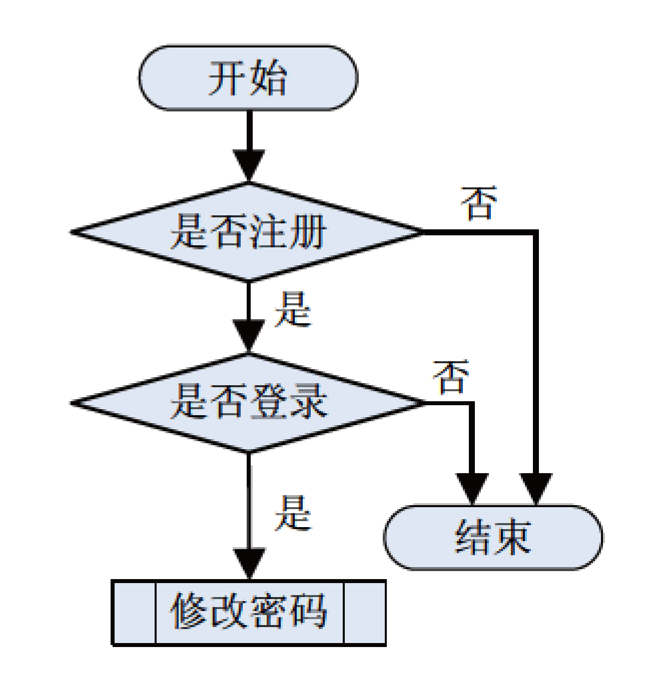


图 4.19　考生信息模块的系统流程图

4.6.2　考生信息模块的技术分析

在考生注册信息模块中应用到 Ajax 无刷新技术获取考生的准考证号码和密码。在介绍该功能的实现过程之前，先对 Ajax 技术中 XmlHttpRequest 对象的属性和方法进行详细讲解。

XmlHttpRequest 对象是 Ajax 技术的核心，有关该对象的属性和方法的详细介绍如表 4.2 和表 4.3 所示。

表 4.2　XmlHttpRequest 对象的属性

属　　性	描　　述
readystate	返回当前的请求状态
onreadystatechange	当 readyState 属性改变时就可以读取此属性值
status	返回 Http 状态码
responseText	将返回的响应信息用字符串表示
ResponseBody	返回响应信息正文，格式为字节数组
ResponseXML	将响应的 domcoment 对象解析成 XML 文档并返回

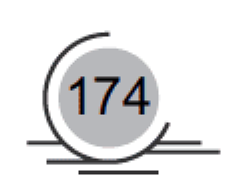




表 4.3　XmlHttpRequest 对象的方法

方　法	描　述
open()	初始化一个新请求
send()	发送请求
GetAllReponseHeaders()	返回所有 Http 头信息
GetResponseHearder()	返回指定的 Http 头信息
SetRequestHeader()	添加指定的 Http 头信息
Abort()	停止当前的 Http 请求

下面对 XmlHttpRequest 对象的常用属性和方法进行详细介绍。

1．readystate 属性

readystate 属性用于返回当前的请求状态，请求状态共有 5 种，如表 4.4 所示。

表 4.4　readystate 属性值

属 性 值	描　述
0	表示尚未初始化，即未调用 open 方法
1	建立请求，但还未调用 send 方法发送请求
2	发送请求
3	处理请求
4	完成响应，返回数据

2．status 属性

status 属性用于返回 Http 状态码，常用 Http 状态码如表 4.5 所示。

表 4.5　Http 状态码

属 性 名	描　述
200	操作成功
404	没有发现文件
500	服务器内部错误
505	服务器不支持或拒绝请求中指定的 Http 版本

3．responseText 属性

responseText 属性将返回的响应信息用字符串来表示。在默认情况下，返回的响应信息的编码格式为 utf-8。

4．responseXML 属性

responseXML 属性用于将响应的 domcoment 对象解析成 XML 文档并返回。

5．open 方法

open 方法用于初始化一个新的请求。

语法：

```
open(String method, String url, Boolean asyn, String user, String password)
```

其中，method 和 url 是必选参数，asyn、user 和 password 是可选参数。open()方法各参数说明如表 4.6 所示。

表 4.6　open()方法参数

参数名称	描　　述
method	指明了新请求的调用方法，其取值有 get 和 post
url	表示要请求页面的 URL 地址。格式可以是相对路径、绝对路径或网络路径
asyn	说明该请求是异步传输还是同步传输，默认值为 true（允许异步传输）
user	服务器验证时的用户名
password	服务器验证时的密码

6．send()方法

send()方法用于发送请求到服务器。

语法：

```
send(body)
```

如果没有要发送的内容，则 body 可以省略或为 Null。

7．GetAllReponseHeaders()方法

GetAllReponseHeaders()方法用于获取响应的所有 Http 头信息。获取到的信息是按照“名称/键值”排列的，信息之间是用“：”冒号进行分隔的。

语法：

```
GetAllReponseHeaders()
```

8．GetResponseHeader()方法

GetResponseHeader()方法是获取响应中指定的 Http 头信息。

语法：

```
GetResponseHeader(String head)
```

9．SetRequestHeader()方法

SetRequestHeader()方法用于添加一个 Http 头信息。

语法：

```
SetRequestHeader(name,value)
```

其中，name 表示 Http 头名称，value 表示信息内容。



10．Abort()方法

Abort()方法用于取消一个请求。

语法：

```
Abort()
```

4.6.3　考生注册的实现过程

考生注册使用的数据表：tb_user

在网络在线考试系统首页，单击“考生注册”超链接，即可进入考生注册页面，输入个人的真实资料后，单击“注册”按钮，系统将根据输入的个人资料自动生成准考证号和考生密码，如图 4.20 所示。



图 4.20　考生注册页面的运行结果

说明 考生需要通过准考证号和密码登录网络在线考试的核心模块，即在线考试及成绩查询等。

在考生注册页面 zhuce.php 页中实现考生注册信息的提交，并且通过 Ajax 的页面无刷新技术获取考生的准考证号和密码。zhuce.php 页中的关键代码如下：

例程02　代码位置：光盘\TM\04\online_ks\zhuce.php

```
<!--调用一个 JavaScript 脚本文件-->
<script type="text/javascript" src="js/zhuce_js.js"></script>
<!--通过 onClick 事件调用一个自定义函数-->
<input type="button" name="Submit" value="注册" onClick="process()">
<!--输出一个 div 标签中的值-->
<div id="divMessage" />
```

在 zhuce_js.js 文件中首先定义一个 createXmlHttpRequestObject()对象，并获取 XMLHttpRrequest 对象，然后定义 xmlHttp 用来存储将要使用的 XMLHttpRrequest 对象。关键代码如下：

例程03　代码位置：光盘\TM\04\online_ks\js\zhuce_js.js

```
var xmlHttp=createXmlHttpRequestObject();                    //定义 XMLHttpRrequest 对象
function createXmlHttpRequestObject(){                       //获取 XMLHttpRrequest 对象
```

```
    var xmlHttp;                                   //用来存储将要使用的 XMLHttpRrequest 对象
    if(window.ActiveXObject){                      //如果在 internet Explorer 下运行
        try{
            xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
        }catch(e){
            xmlHttp=false;
        }
    }else{                                         //如果在 Mozilla 或其他的浏览器下运行
        try{
            xmlHttp=new XMLHttpRequest();
        }catch(e){
            xmlHttp=false;
        }
    }
    if(!xmlHttp)                                   //返回创建的对象或显示错误信息
        alert("返回创建的对象或显示错误信息");
        else
        return xmlHttp;
}
```

然后使用 XMLHttpRequest 对象创建异步 Http 请求，定义函数 process()，对表单中提交的数据进行判断，并且获取表单中输入的信息，在服务器端执行 zhuce_ok.php 文件，向服务器发送请求。关键代码如下：

例程04　代码位置：光盘\TM\04\online_ks\js\zhuce_js.js

```
function process(){                                //使用 XMLHttpRequest 对象创建异步 Http 请求
    if(form1.username.value==""){                  //如果考生姓名为空
        alert("请输入姓名！");                      //则弹出提示
            form1.username.select();               //返回焦点到考生姓名文本框
            return(false);
        }
    if(form1.tel.value==""){                       //如果考生联系电话为空
        alert("请输入电话号码！");                  //则弹出提示
        form1.tel.select();                        //返回焦点到考生联系方式文本框
        return(false);
        }
    if(checkphone(form1.tel.value)!=true){         //如果考生联系方式格式不正确
        alert("您输入的电话号码的格式不正确！");     //则弹出提示
        form1.tel.select();                        //返回焦点到考生联系方式文本框
        return(false);
        }
    if(form1.address.value==""){                   //如果考生联系地址为空
        alert("请输入联系地址！");                  //则弹出提示
        form1.address.select();                    //返回焦点到考生联系地址文本框
        return(false);
        }
    if(xmlHttp.readyState==4 || xmlHttp.readyState==0){    //在 xmlHttp 对象不忙时进行处理
    names = document.getElementById("username").value;      //获取用户在线表单中输入的姓名
    tels = document.getElementById("tel").value;            //获取用户在线表单中输入的电话
    addresss =document.getElementById("address").value;     //获取用户在线表单中输入的地址
```

```
        xmlHttp.open("GET","zhuce_ok.php?online_user="+names+"& online_tel="+tels+"&
online_address="+addresss,true);                                    //在服务器端执行 zhuce_ok.php
        xmlHttp.onreadystatechange=handleServerResponse;            //定义获取服务器端响应的方法

        xmlHttp.send(null);                                         //向服务器发送请求
        }else
        setTimeout('process()',1000);                               //如果服务器忙，1 秒后重试
}
```

最后获取从服务器端返回的消息。关键代码如下：

例程05　代码位置：光盘\TM\04\online_ks\js\zhuce_js.js

```
//当收到服务器端的消息时自动执行
function handleServerResponse(){
        if(xmlHttp.readystate==4){                                  //在处理结束时进入下一步
                if(xmlHttp.status==200){                            //状态为 200 表示处理成功结束
                        xmlResponse=xmlHttp.responseXML;            //获取服务器端发来的 XML 信息
                        xmlDocumentElement=xmlResponse.documentElement; //获取 XML 中的文档对象（根对象）
                        helloMessage=xmlDocumentElement.firstChild.data;    //获取第一个文档子元素的文本信息
                        //使用从服务器端发来的消息更新客户端显示的内容
                        document.getElementById("divMessage").innerHTML='<i>'+helloMessage+'</i>';
                        setTimeout('process()',1000);               //重新开始
                }else{
                        //如果 Http 的状态不是 200 表示发生错误
                alert("There was a problem accessing the server:"+xmlHttp.statusText);
                }
        }
}
```

zhuce_ok.php 文件在服务器端被执行，将获取到的用户信息和随机生成的准考证号、密码添加到数据库中，并且将准考证号和密码返回到考生注册页面中。关键代码如下：

例程06　代码位置：光盘\TM\04\online_ks\zhuce_ok.php

```
<?php
header('Content-Type: text/xml');                                   //创建一个 XML 格式输出
echo '<?xml version="1.0" encoding="gb2312" standalone="yes" ?>';   //创建 XML 头
echo '<response>';                                                  //创建<response>元素
❶    echo $online_user=$_GET[online_user];                          //获取考生姓名
$online_tel=$_GET[online_tel];                                      //获取考生联系电话
$online_address=$_GET[online_address];                              //获取考生联系地址
❷    $online_number=substr(mt_rand(100000,999999),0,6);             //生成 6 位随机的准考证号码
$online_pass=substr(mt_rand(100000,999999),0,6);                    //生成 6 位随机的考生密码
/* ****************************根据从客户端获取的用户创建输出****************************** */
include("conn/conn.php");
$query=mssql_query("insert into tb_user(online_user,online_tel,online_address,online_number,online_pass)
values('$online_user','$online_tel','$online_address','$online_number','$online_pass')");
if($query==true){
        echo $online_user;
        echo "用户注册成功，这是您的准考证号码$online_number.和密码$online_pass.";
```

```
}
    echo '</response>';                              //关闭<response>元素
?>
```

代码贴士

❶ $_GET 预定义变量：PHP 使用$_GET 预定义变量自动保存通过 get 方法传过来的值，使用格式为：

```
$_GET[name]
```

❷ substr()函数：从指定的字符串中按照指定的位置截取一定长度的字符。语法如下：

```
string substr(string str,int start,int length)
```

参数 str 指定字符串对象。参数 start 指定开始截取字符串的位置，如果为负数，则从字符串的末尾开始截取。参数 length 是可选参数，指定截取字符的个数，如果为负数，则表示取到倒数第 length 个字符。

4.6.4 单元测试

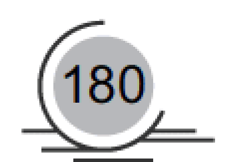
在应用 Ajax 技术实现在线考试的用户注册模块后，为保证程序能够正常运行，必须对其进行测试。考生注册模块的运行结果如图 4.21 所示。



图 4.21　考生注册模块运行出现的错误

从弹出的错误信息来看，是程序中与 JavaScript 脚本有关的内容出现了问题。经过反复测试和查找发现，问题不是出现在 zhuce_js.js 文件中，因为在该文件中定义的自定义函数 process()是可以运行的，所以问题应该是出现在对服务器返回的信息进行输出中，即 zhuce_ok.php 文件中。对该文件进行分析发现，原来在应用 Ajax 技术时，服务器端返回的消息采用的是 XML 格式，可是在客户端接收 XML 文档时却忽略了“header('Content-Type: text/xml');”头部信息的使用，从而导致客户端用于解析 XML 的 API 抛出一个错误信息。只有在设置头部信息（header('Content-Type: text/xml');）后，客户端才可以正确地接收到服务器端传递的 XML 文档信息。正确的程序代码如下：

```
<?php
//创建一个 XML 格式输出
header('Content-Type: text/xml');            //标记输出的是 XML 文档
//创建 XML 头
echo '<?xml version="1.0" encoding="gb2312" standalone="yes" ?>';
?>
```



4.7　在线考试模块设计

4.7.1　在线考试模块的概述

在线考试模块的主要功能是允许考生在网站上针对指定的课程进行考试。在该模块中，考生首先需要登录到本系统中，阅读考试规则，在同意所列出的考试规则后，才能选择考试类别和套题，然后进入考试页面进行答题，当考生提交试卷或者到达考试结束时间时，系统将自动对考生提交的试卷进行评分，并给出最终考试成绩。在线考试模块的系统流程如图 4.22 所示。

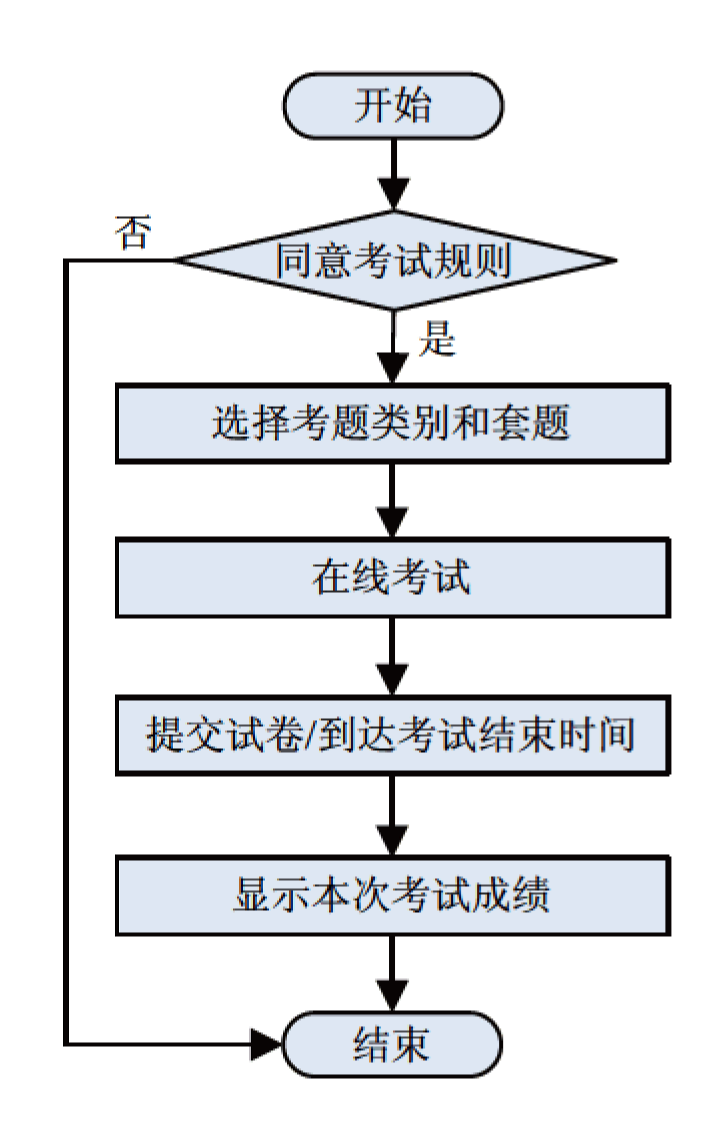


图 4.22　在线考试模块的系统流程图

4.7.2　在线考试模块的技术分析

在线考试模块中最核心的功能是如何输出考题、如何对提交的答案进行判断，并且将答案输出到当前页中。由于上述 3 个方面的内容都是在同一页面中显示，所以在技术实现过程中这 3 个方面的内容是以一个整体形式出现的。这里以单选题为例进行讲解。

首先以上一页表单中提交的值为条件，执行查询语句，通过 while 循环语句输出查询结果；然后根据输出的查询结果创建单选按钮组，生成考题；最后，将考题答案提交到本页，对提交的答案与数据库中存储的正确答案进行比较，并且将答案输出到本页中。单选题输出、答案判断和考题答案输出的关键代码如下：

例程07　代码位置：光盘\TM\04\online_ks\ksks.php

```
<?php
/*根据提交的变量执行查询语句，从数据库中读取数据，指定考题的类型为 0，表示为单选题*/
$query0=mssql_query("select * from tb_kt where kt_lb='$kt_lbes' and kt_lx='0' and kt_small_lb='$kt_small_lb'");
$x=1;                                          //定义变量$x=1，用于输出题号
$fen0=0;                                       //定义变量$fen0=0，用于考试分数的计算
while($myrow0=mssql_fetch_array($query0)){     //执行 while 循环语句
?>
<tr>
    <!--输出单选题的题号、考题内容以及该题的分数-->
    <td width="443" height="20" bgcolor="#FFFFFF">  <?php echo $x.".".$myrow0["kt_nr"]?> </td>
    <td bgcolor="#FFFFFF"><?php echo $myrow0[kt_fs];?></td>
    <!------------------------------------------------>
</tr>
<?php
    /*****获取考题提交的答案，将提交的考题答案进行分割，并赋给变量$str0*******/
    $array0=explode("*",$myrow0["kt_daan"]);   //将数据库中的考题答案以*进行分割，并写入到数组中
    if($_POST[Submit]!=""){                    //判断表单提交是否为空
        for($a=0;$a<count($array0);$a++){      //循环输出数组中的答案
```

```
                    if($array0[$a]!=""){                              //数组中的值不为空
                        if($array0[$a]==$_POST[$myrow0[kt_id]]) {//如果数组中的值与表单提交的值相等
                            $str0=$_POST[$myrow0[kt_id]];         //将表单提交的值赋给变量$str0
                        }
                    }
                }
            }
        /****************************************************************/
        for($a=0;$a<count($array0);$a++){                  //执行 for 循环语句，根据考题答案来决定循环的次数
          if($array0[$a]!=""){                             //判断如果考题答案不为空，则执行下面的内容
?>
<tr>
      <td height="20" bgcolor="#FFFFFF">  
          <!--创建单选按钮组，用于提交答案-->
          <input type="radio" name="<?php echo $myrow0[kt_id];?>" value="<?php echo $array0[$a];?>">
          <?php echo $array0[$a];?>
          <!--------------------------------->
      </td>
    <td bgcolor="#FFFFFF">  
      <?php
      /* ***********************判断提交的答案是否正确，并且输出答案************************** */
          if($_POST[$myrow0[kt_id]]==true){                //判断表单提交的值是否为空
              if($a==0)                                    //如果变量$a==0，则输出下面的内容
              if($myrow0["kt_zqdaan"]==$str0){             //判断提交的答案是否与数据库中的正确答案相同
                  //如果提交的答案正确，则输出下面的内容
                  echo "您输入的答案 ";
                  echo "<font color='#FF0000'>".substr($str0,0,1)."</font>";        //获取答案的选项
                  echo " 正确  分数:";
                  echo "<font color='#FF0000'>".$myrow0[kt_fs]."</font>";           //输出考题的分数
                  $fen0+=$myrow0["kt_fs"];                                          //对考题的分数进行核算
              }else{                                       //如果提交的答案不正确，则输出下面的内容
                  echo "您输入的答案 ";
                      echo "<font color='#FF0000'>".substr($str0,0,1)."</font>";
                      echo " 错误  ";
                  echo "正确答案: <font color='#FF0000'>".substr($myrow0[kt_zqdaan],0,1)."</font>" ;
              }
          }
      ?></td>
<?php
              }
          }   $x++;                                        //实现考题题号的循环
      }    ?>
```

4.7.3 应用 Ajax 在线答题的实现过程

在线答题使用的数据表：tb_kt

在线答题是本项目中最核心的技术，也是本系统开发的最终目的。在线答题模块的主要功能是实现在线考试，在当前页面中输出考题答案和考试成绩，并且采用计时控制考试时间，如果考试超过考

试时间，则系统自动提交考题答案，并给出考试成绩。

考试时间计时和剩余时间的计算是应用 Ajax 技术来实现的，有关该技术的实现方法可以参考 4.10.2 节，这里不再讲解。

在线答题技术中对单选题的处理方法已经在技术分析中进行了详细的讲解，这里就不再赘述，下面讲解在线答题中多选题的处理方法，其实现的原理与单选题是相同的，只是在输出考题的答案时使用的是复选框，不再是单选按钮组，并且在对复选框名称进行设置时增加一个变量来对不同考题的答案进行区分。首先以上一页表单中提交的值为条件，执行查询语句，通过 while 语句输出查询结果；然后根据输出的查询结果创建复选框，生成考题；最后，将考题答案提交到本页，对提交的答案与数据库中存储的正确答案进行比较，并且将答案输出到本页中。

首先从数据库中读取数据，定义输出考题时应用到的变量值，并且输出考题的编号和内容。关键代码如下：

例程08　代码位置：光盘\TM\04\online_ks\ksks.php

```
<?php
      //根据表单提交的值执行查询语句，从数据库中读取符合条件的数据
      $query1=mssql_query("select * from tb_kt where kt_lb='$kt_lbes' and kt_lx='1' and kt_small_lb='$kt_small_lb'");
      $y=1;                                              //定义变量$y 用于生成考题编号
      $fen1=0;                                           //定义变量$fen1 用于分数的统计
      $ii=0;                                             //定义变量$ii 用于控制考题答案输出的次数
      while($myrow1=mssql_fetch_array($query1)){         //循环输出数据
?>
<!--- -----------------------------------输出考题编号和考题的内容-------------------------------------- -->
<tr>
     <td height="20" bgcolor="#FFFFFF">  <?php echo $y."."$myrow1["kt_nr"]?> </td>
     <td bgcolor="#FFFFFF"><?php echo $myrow1[kt_fs];?></td>
</tr>
<!-- ----------------------------------------------------------------------------------------------- --->
```

然后，对表单中提交的答案进行处理，与数据库中存储的正确答案相比较，如果答案正确，则将表单提交的答案写入到一个空数组中。代码如下：

例程09　代码位置：光盘\TM\04\online_ks\ksks.php

```
<?php
      /*****************对考题提交的答案进行处理，输出考题答案********************/
      $array_a1=array();                                 //定义一个空数组
      //将数据库中考题的答案以*进行分割，存储到数组$array1 中
❶     $array1=explode("*",$myrow1["kt_daan"]);
     if($_POST[Submit]!=""){                             //判断表单提交的值是否为空
           for($i=0;$i<count($array1);$i++){             //循环读取数组$array1 中的考题答案
                 if($array1[$i]!=""){                    //判断数组中的答案是否为空
                       //对表单中提交的答案与数据库中的答案进行比较
                       if($array1[$i]==$_POST[$myrow1[kt_id]."-".$i]) {
                             //如果答案相同，则将表单提交的答案写入到空数组$array_a1 中
❷                            array_push($array_a1,$_POST[$myrow1[kt_id]."-".$i]);
                       }
                 }
           }
```

```
    }
?>
```

代码贴士

❶ explode()函数：将字符串依指定的字符串或字符 separator 切开。语法如下：

```
array explode(string separator, string string, [int limit])
```

该函数返回由字符串组成的数组，每个元素都是 string 的一个子串，它们被字符串 separator 作为边界点分隔出来。如果设置了 limit 参数，则返回的数组包含最多 limit 个元素，而最后那个元素将包含 string 的剩余部分。如果 separator 为空字符串（""），explode()函数将返回 false；如果 separator 所包含的值在 string 中找不到，那么 explode()函数将返回包含 string 单个元素的数组。如果参数 limit 是负数，则返回除了最后的-limit 个元素外的所有元素。

❷ array_push()函数：将数组当成一个栈，并将传入的变量压入该数组的末尾。该数组的长度将增加入栈变量的数目。返回数组新的单元总数。语法如下：

```
int array_push ( array array, mixed var [, mixed ...])
```

参数 array 指输入的数组；参数 var 是必要参数，用来压入数组。

接着创建考题提交的复选框，根据考题的 ID 和变量$I 定义复选框的名称。代码如下：

例程10　代码位置：光盘\TM\04\online_ks\ksks.php

```
<?php
    $str1=implode("*",$array_a1);                    //以*来分割数组$array_a1 中的数据，返回值为变量$str1
    for($i=0;$i<count($array1);$i++){                //根据数据库中存储的答案执行循环语句
    if($array1[$i]!=""){                             //判断答案不为空
?>
<tr><td height="20" bgcolor="#FFFFFF">  
    <!-- -------------------创建答案提交的复选框，并且设置复选框的名称为考题的 id 和变量$i----------------- --->
    <input type='checkbox' name='<?php echo $myrow1[kt_id]."-".$i?>' value='<?php echo $array1[$i];?>'>
<?php echo $array1[$i];?>
    <!-- ------------------------------------------------------------------------------------------------------------- -->
</td><td bgcolor="#FFFFFF"> 
```

再次对提交的答案进行判断，其中主要应用的是 explode()、list()和 substr()函数来实现答案的判断和输出。程序代码如下：

例程11　代码位置：光盘\TM\04\online_ks\ksks.php

```
<?php
    /* ********************对提交的答案进行判断，并且输出答案和最终得分*************************/
    if($_POST[$myrow1[kt_id]."-".$i]==true){         //判断提交的答案是否为真
        if($ii==0){                                  //判断变量$ii 的值是否为 0，控制答案输出的次数
            if($myrow1["kt_zqdaan"]==$str1){         //判断表单提交的答案是否与数据库中的正确答案相等
                /* **************如果答案正确，则输出提交的答案，并且给出该题的得分************** */
                echo "您输入的答案 ";
                $arrayesg=explode("*",$str1);                                //以*来分隔提交的答案
❶               for($gg=0;$gg<count($arrayesg);$gg++){                       //循环读取答案的内容
❷                   list($name,$valuesg)=each($arrayesg);                    //输出答案
                    echo "<font color='#FF0000'>".substr($valuesg,0,1)."</font>";   //截取答案的内容
                }
```



```
                    echo " 正确  分数:";
                    echo "<font color='#FF0000'>".$myrow1[kt_fs]."</font>";
                    $fen1+=$myrow1["kt_fs"];                    //统计考题的分数
            }else{
                /* ***********如果输出的答案错误，则提示您输入的答案错误，并给出正确的答案*********** */
                    echo "您输入的答案 ";
                    $arrayes=explode("*",$str1);
                    for($g=0;$g<count($arrayes);$g++){
                        list($name,$values)=each($arrayes);
                        echo "<font color='#FF0000'>".substr($values,0,1)."</font>";
                    }
                    echo " 错误  正确答案: ";
                    $arrayes=explode("*",$myrow1[kt_zqdaan]);
                    for($g=0;$g<count($arrayes);$g++){
                        list($name,$values)=each($arrayes);
                        echo "<font color='#FF0000'>".substr($values,0,1)."</font>";
                    }
                }
            }$ii=1;                                         //控制每道题的答案只输出一次数
        }
    ?></td></tr>
<?php }   }
        $y++;                                               //控制每道题的题号
        }
?>
```

代码贴士

❶ count()函数：返回数组中的单元数目，用来计算数组中值的个数。语法如下：

```
int count( mixed array [, int mode])
```

参数 array 是必要参数，指输入的数组；参数 mode 是可选参数，默认值是 0。

❷ list()函数：把数组中的值赋给一些变量。与 array()函数类似，这不是真正的函数，而是语言结构。list()函数仅能用于数字索引的数组，且数字索引从 0 开始。语法如下：

```
void list( mixed ...)
```

参数 mixed 为被赋值的变量名称。

each()函数用于返回数组中当前指针位置的键名和对应的值，并向前移动数组指针。

```
array each( array array)
```

参数 array 为输入的数组。

4.7.4　分数统计和成绩保存的实现过程

分数统计和成绩保存使用的数据表：tb_user

在线考试模块中不但完成考试的答题和判卷的操作，而且可以直接对考试的分数进行统计，获取

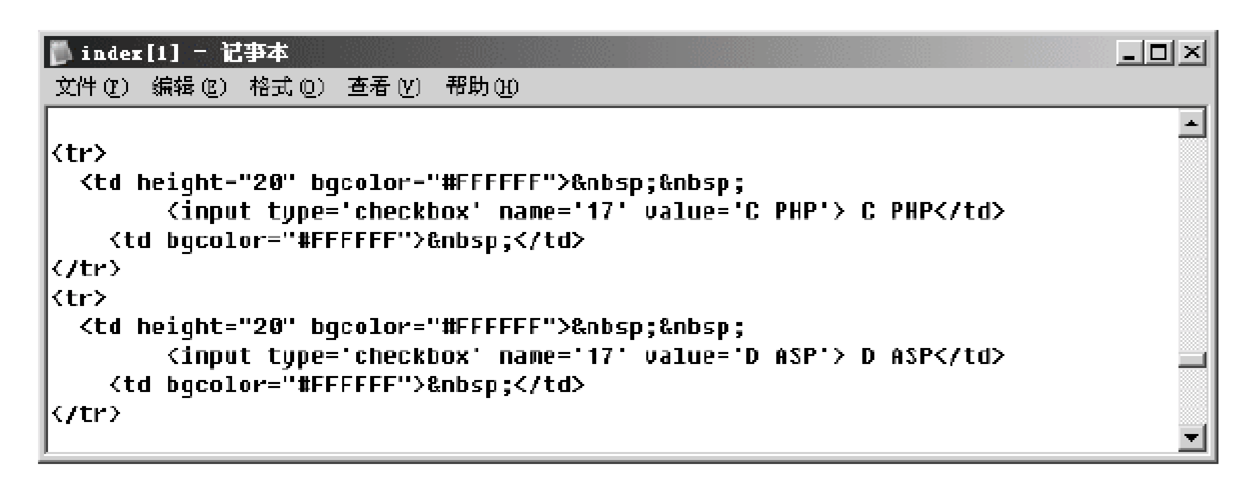
考生的最终得分，将考生的成绩保存到数据库中，并且更改数据库中考生的考试信息，说明该考生已经完成本次考试，不可以再进行本类别的考试。考试分数统计和存储考试成绩的关键代码如下：

例程12　代码位置：光盘\TM\04\online_ks\ksks.php

```
<?php
    /* ****************统计考生的考试成绩，其中的变量分别代表考生各个题型的得分***************** */
    $zf=$fen0+$fen1+$fen2+$fen3;                    //考试题分数统计
    echo "您的总成绩是:";                            //输出字符串
    echo $zf;                                       //输出考试成绩
?>
<?php
/* **********************************将考生的成绩存储到数据库中********************************* */
if($Submit=="提交"){                                //首先判断考生是否提交考题的答案
$data=date("Y-m-d H:i:s");                          //获取考试的时间
$grade="update tb_user set online_grade='$zf',online_subject='$_POST[kt_ lbes]',online_pt='1',online_date= '$data'
where online_number='$_SESSION[online_number]'";
$grade_result=mssql_query($grade);
}
?>
```

4.7.5　单元测试

在完成在线考试模块的设计后对该模块进行测试，发现在执行多选题提交的过程中，没有任何值返回。但是在执行单选题提交时可以正常运行，查看该文件的源文件发现，原来错误出现在复选框名称的设置上，所有复选框都是相同的值，其运行结果如图 4.23 所示。



```
<tr>
  <td height-"20" bgcolor-"#FFFFFF">  
        <input type='checkbox' name='17' value='C PHP'> C PHP</td>
    <td bgcolor="#FFFFFF"> </td>
</tr>
<tr>
  <td height="20" bgcolor="#FFFFFF">  
        <input type='checkbox' name='17' value='D ASP'> D ASP</td>
    <td bgcolor="#FFFFFF"> </td>
</tr>
```

图 4.23　查看源文件的结果

在设置复选框的名称时必须为每一个题设置不同的复选框名称，否则将不能获取到表单提交的数据，这就是多选题与单选题进行输出和判断的主要区别，多选题中有多个答案，而单选题只有一个，所以单选题的单选按钮组的名称可以直接使用考题的 ID 进行命名，但是多选题必须在考题 ID 的基础之上再定义一个变量来区分每个复选框的名称。在程序中，错误的复选框名称定义的代码如下：

```
<input type="radio" name="<?php echo $myrow0[kt_id];?>" value="<?php echo $array0[$a];?>">
```

正确的复选框名称定义应该是在考题 ID 的基础之上加一个变量$i，程序代码如下：

```
<input type='checkbox' name='<?php echo $myrow1[kt_id]."-".$i?>' value='<?php echo $array1[$i];?>'>
```

对程序代码进行修改，重新保存，程序可以正常运行。



4.8 后台首页设计

4.8.1 后台首页概述

网络在线考试系统的后台首页是管理员对网站信息进行管理的首页面。在该页面中，管理员可以清楚地了解网站后台管理系统包含的基本操作。网络在线考试网后台首页包含的主要模块介绍如下。

- ☑ 管理员信息管理：主要用于修改管理员信息。
- ☑ 考生信息管理：主要包括查看注册考生信息列表和考生信息查询、考试成绩查询、删除已注册的考生信息。
- ☑ 考题类别管理：主要包括查看考题类别列表、添加考题类别信息和删除考题类别信息。
- ☑ 考题信息添加：主要用于为各类套题添加单选题、多选题、问答题和论述题，并设置每题的分数及内容。
- ☑ 考题信息管理：主要包括查看考题类别列表、修改套题信息和删除套题信息。
- ☑ 退出管理：主要用于退出后台管理系统。

下面看一下本案例中提供的后台首页，该页面在本书光盘中的路径为\TM\04\online_ks\admin\index.php，如图 4.24 所示。



图 4.24 网络在线考试系统的后台首页

4.8.2 后台首页技术分析

网络在线考试后台主要应用 switch 语句实现了一种类似于框架的网页嵌套技术。为了能够更好地理解这个技术，下面进行详细介绍。

switch 语句一行接一行地执行（实际上是语句接语句），开始时没有代码被执行。仅当一个 case 语句中的值和 switch 表达式的值匹配时，PHP 才开始执行语句，直到 switch 的程序段结束或者遇到第一个 break 语句为止。如果不在 case 的语句段最后写上 break，PHP 将继续执行下一个 case 中的语句段。

switch 语句的语法如下：

```
switch(expr) {
      case value : statement; break;
        …
      default : statement; break;
}
```

switch 语句的参数说明如表 4.7 所示。

表 4.7　switch 语句的参数说明

参　数	说　明
expr	必要参数。定义判断的条件表达式
statement	必要参数。满足条件后，循环执行的语句

4.8.3　后台首页的实现过程

网络在线考试后台主要应用 switch 语句和 include 包含语句实现了类似于框架的网页嵌套技术。其中，应用 switch 语句，根据变量标识$htgl 提交的值进行判断，应用 include 包含语句调用不同功能模块的脚本文件。代码如下：

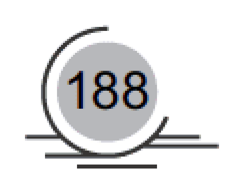
例程13　代码位置：光盘\TM\04\online_ks\admin\index.php

```
<?php
switch($htgl){                                    //判断后台超链接名称
      case "考生信息管理":                        //如果名称为“考生信息管理”
            include("ksxx_gl.php");               //则应用 include()语句调用 ksxx_gl.php 文件
            break;                                //退出语句段的执行
      case "考题类别管理":                        //如果名称为“考题类别管理”
            include("ktlb_gl.php");               //则应用 include()语句调用 ktlb_gl.php 文件
            break;                                //退出语句段的执行
      case "考题信息添加":                        //如果名称为“考题信息添加”
            include("ktxx_tj.php");               //则应用 include()语句调用 ktxx_tj.php 文件
            break;                                //退出语句段的执行
      case "考题信息管理":                        //如果名称为“考题信息管理”
            include("ktxx_gl.php");               //则应用 include()语句调用 ktxx_gl.php 文件
            break;                                //退出语句段的执行
}
?>
```

4.9　考题信息管理模块设计

4.9.1　考题信息管理模块的概述

考题信息管理模块主要包括查询考题信息、添加考题信息、修改考题信息和删除考题信息等 4 个功能。考题信息管理模块的框架如图 4.25 所示。





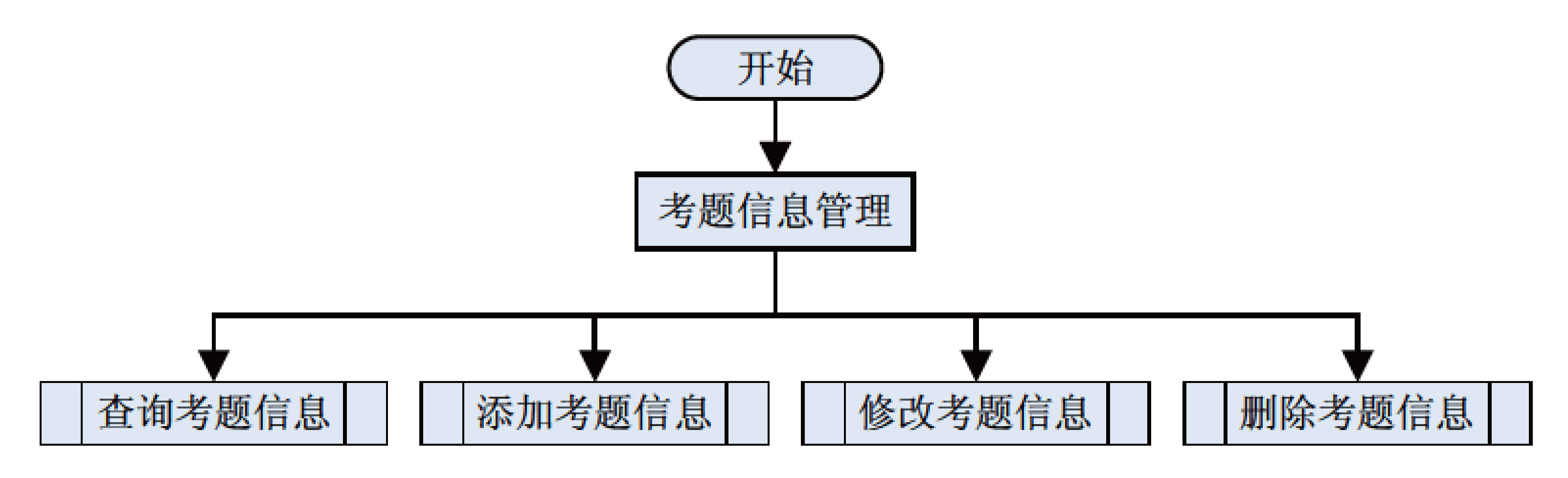


图 4.25　考题信息管理模块的框架图

4.9.2　考题信息管理模块的技术分析

在实现考题信息管理模块时，为了更好地管理，因此把考题类别单独存储于一个数据表 tb_lb 中，这样在录入考题信息时就可以把考题类别以下拉列表的形式从数据库中读取出来。这种从下拉列表中动态显示数据表某列的字段值的方法，不仅方便管理员更快捷、灵活地操作网络在线考试系统，更大大提高了工作效率，达到事半功倍的效果。

下拉列表是一种最节省空间的数据显示方式，正常状态下只能看到一个选项，单击控制按钮后，可以显示一定数量的选项，如果超出这个数量，会自动显示滚动条，管理员可以通过拖动滚动条来选择各选项。

下面介绍从下拉列表中动态显示数据表某列的字段值的方法。

首页，创建与数据库的连接。代码如下：

```
<?php
$id=mssql_connect('localhost','sa','');
mssql_select_db("db_online ",$id);
?>
```

然后应用下拉列表框和 select 查询语句相结合实现在下拉列表中显示数据表中的 online_ktlb 字段的值，通过 while 循环语句进行输出。代码如下：

```
<?php
$query=mssql_query("select * from tb_ktlb");                    //查询考题类别表中的信息
while($myrow=mssql_fetch_array($query)){                        //应用 while 循环语句输出考题信息
?>
<!-- ------------------------------------输出考题类别表中的 online_ktlb 字段--------------------------------- -->
  <option value="<?php echo $myrow[online_ktlb];?>"><?php echo $myrow[online_ktlb];?></option>
<?php
 }                                                              //while 循环语句结束
?>
```

4.9.3　考题信息添加的实现过程

▦　*添加考试题目信息使用的数据表：tb_Lesson、tb_taoTi 和 tb_Questions*

考试题目添加包含两个步骤：一是为添加的考试题目选择专业、课程以及套题；二是将填写的考试题目信息插入到数据库中。考题信息添加页面的运行结果如图 4.26 所示。



图 4.26　考题信息添加页面的运行结果

考题信息添加页面涉及的 HTML 表单的重要元素如表 4.8 所示。

表 4.8　考题信息添加页面涉及的 HTML 表单的重要元素

名　称	类　型	重 要 属 性	含　义
form2	form	method="post" action="ktxx_tj_ok.php"	表单
kt_lb	select	<?php $query=mssql_query("select * from tb_ktlb"); while($myrow=mssql_fetch_array($query)){ ?> <option value="<?php echo $myrow[online_ktlb];?> "> <?php echo $myrow[online_ktlb];?> </option> <?php }?>	考题类别
kt_small_lb	select	<select> <option value="第一套题">第一套题</option> <option value="第二套题">第二套题</option> <option value="第三套题">第三套题</option> <option value="第四套题">第四套题</option> </select>	所属套题
kt_lx	select	<select> <option value="2">简答</option> <option value="3">论述</option> <option value="0">单选</option> <option value="1">多选</option> </select>	考题类型
kt_fs	text	size="10"	考试成绩
kt_nr	textarea	cols="60" rows="5"	考题内容
kt_daan	textarea	cols="60" rows="5"	考题答案



续表

名　称	类　型	重 要 属 性	含　义
kt_zqdaan	textarea	cols="60" rows="5"	考题正确答案
Submit2	submit	value="提交考题"	“提交考题”按钮

添加考试题目首先要选择考题的类别，然后选择所属套题，再选择考题类型，其实现过程在表 4.8 所示表单的重要属性中给出了完整的代码，这里不再赘述，最后根据考题类别将考题内容添加完整。

提交考题表单到数据处理页，程序处理页面首先应用变量获取到表单数据，然后应用 insert…into 语句将其插入到考题信息表 tb_kt 中。如果考题信息添加成功，则弹出提示信息，并重新定位到考题信息添加页面。数据处理页的完整代码如下：

例程14　代码位置：光盘\TM\04\online_ks\admin\ktxx_tj_ok.php

```
<?php
include("../conn/conn.php");                          //连接数据库文件
if($Submit2=="提交考题"){                               //如果单击了“提交考题”按钮，则执行下面语句
      $queryes=mssql_query("insert into tb_kt
(kt_lb,kt_lx,kt_fs,kt_nr,kt_daan,kt_zqdaan,kt_small_lb)values('$kt_lb','$kt_lx','$kt_fs','$kt_nr','$kt_daan','$kt_zqd
aan','$kt_small_lb')");                                //向考题信息表中添加数据
      if($queryes){                                    //如果添加操作成功，则弹出提示，并重定向网页
            echo "<script>alert('考题添加成功！'); window.location.href='index.php?htgl=考题信息添加';</script>";
      }
}
?>
```

4.9.4　查询考题信息的实现过程

查看考试题目列表使用的数据表：tb_kt、tb_ktlb

管理员登录后，单击“考题信息管理”超链接，进入到查询考题信息页面，选择考题类别后，单击“考题查找”按钮，将查询出该类别下的所有考题信息。同时提供修改考题信息和删除考题信息的功能。查询考题信息页面的运行结果如图 4.27 所示。

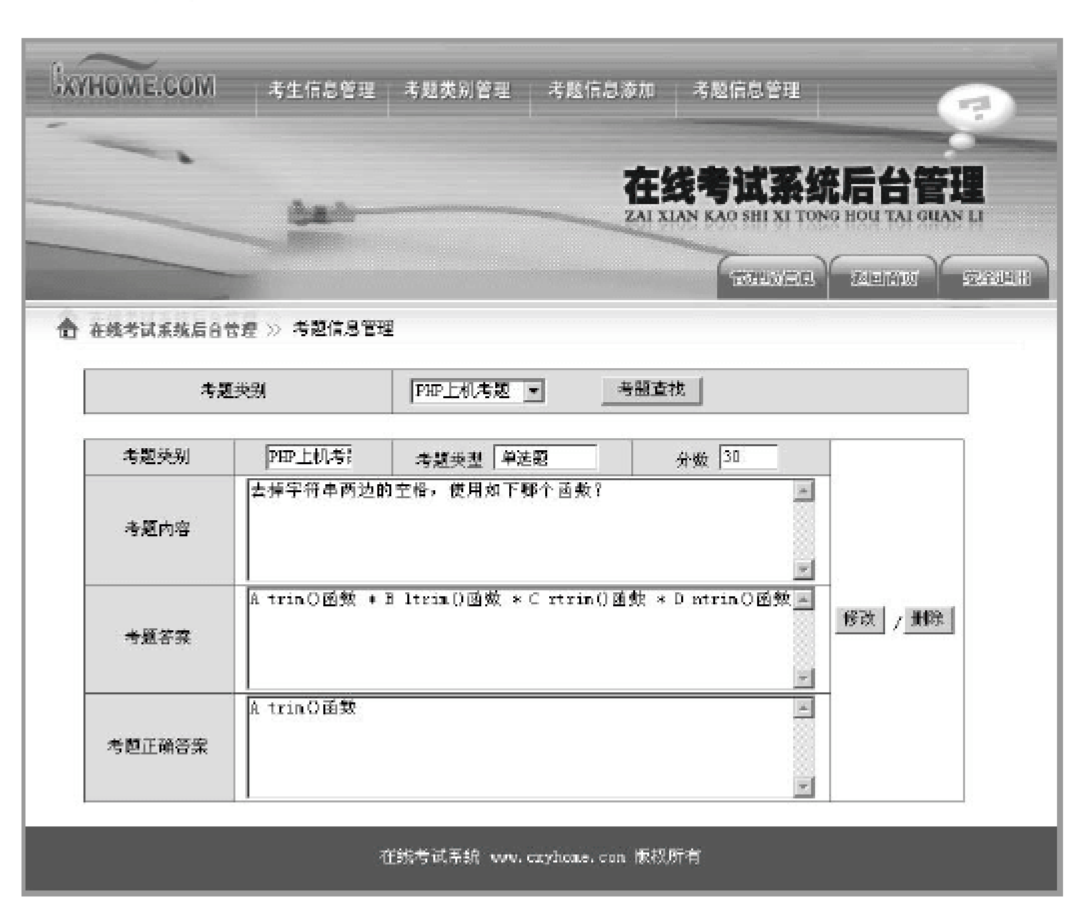


图 4.27　查询考题信息页面的运行结果

查询考题信息页面涉及的 HTML 表单的重要元素如表 4.9 所示。

表 4.9　查询考题信息页面涉及的 HTML 表单的重要元素

名　称	类　型	重 要 属 性	含　义
form1	form	method="post" action="index.php?htgl=考题信息管理"	查询考题表单
kt_lb	select	<select name="kt_lb" id="kt_lb"> <?php $query=mssql_query("select * from tb_ktlb"); while($myrow=mssql_fetch_array($query)){ ?> <option value="<?php echo $myrow[online_ktlb];?>"> <?php echo $myrow[online_ktlb];?></option> <?php }?> </select>	考题类别
Submit	submit	value="考题查找"	“考题查找”按钮
form2	form	method="post" action="ktxx_gl_ok.php"	考题信息表单
kt_lb	text	value="<?php echo $myrow[kt_lb];?>"	考题类别显示
kt_lx	text	value="<?php echo $myrow[kt_lx];?>"	考题类型
kt_fs	text	value="<?php echo $myrow[kt_fs];?>"	考试成绩
kt_id	hidden	value="<?php echo $myrow[kt_id]?>"	考题类别 id
Submit2	submit	value="修改"	“修改”按钮
Submit3	submit	value="删除"	“删除”按钮
kt_nr	textarea	<?php echo $myrow[kt_nr];?>	考题内容
kt_daan	textarea	<?php echo $myrow[kt_daan];?>	考题答案
kt_zqdaan	textarea	<?php echo $myrow[kt_zqdaan];?>	考题正确答案

在考试题目查询页面中，首先建立用于查询的表单 form1，该表单中包含“考题类别”列表/菜单控件以及“考题查找”按钮。

当管理员选择考题类别后，单击“考题查找”按钮，提交考题类别 kt_lb 到当前页，然后根据获取到的考题类别检索考题信息表 tb_kt，并将该类别下的所有考题信息输出到浏览器。代码如下：

例程15　代码位置：光盘\TM\04\online_ks\admin\ktxx_gl.php

```
<!--- ---------------------------------------------------考题信息显示---------------------------------------------------- -->
<table width="682" height="168" border="0" cellpadding="0" cellspacing="1" bgcolor="#5D554A">
<?php
include("../conn/conn.php");                                          //连接数据库文件
❶    $query=mssql_query("select * from tb_kt where kt_lb='$kt_lb'");   //根据选择考题类别检索数据信息
❷    while($myrow=mssql_fetch_array($query)){                          //将查询结果应用 while 循环输出
?>
<form name="form2" method="post" action="ktxx_gl_ok.php">
  <tr>
    <!-- ---------------------------------------------考题类别显示------------------------------------------ -->
    <td width="112" height="27" align="center" bgcolor="#DDDDDD">考题类别</td>
    <td width="117" align="center" bgcolor="#DDDDDD">
        <input name="kt_lb" type="text" value="<?php echo $myrow[kt_lb];?>" size="8">
    </td>
```



```
    <!-- ----------------------------------------------考题类型显示------------------------------------------- -->
    <td width="180" align="center" bgcolor="#DDDDDD">考题类型
        <input name="kt_lx" type="text" value="<?php echo $myrow[kt_lx];?>" size="10">
     </td>
    <!-- ----------------------------------------------考试成绩显示------------------------------------------- -->
    <td width="148" align="center" bgcolor="#DDDDDD">分数
        <input name="kt_fs" type="text" value="<?php echo $myrow[kt_fs];?>" size="5">
    </td>
    <!-- ---------------------------建立考题 ID 隐藏域，添加“修改”、“删除”按钮--------------------------- -->
    <td width="99" rowspan="4" align="center" bgcolor="#FFFFFF">
        <input type="hidden" name="kt_id" value="<?php echo $myrow[kt_id]?>">
        <input type="submit" name="Submit2" value="修改">/<input type="submit" name="Submit3" value="删除">
     </td>
  </tr>
    <!-- ----------------------------------------------考题内容显示------------------------------------------- -->
  <tr>
    <td height="43" align="center" bgcolor="#DDDDDD">考题内容</td>
    <td colspan="3" align="center" bgcolor="#FFFFFF"><textarea name="kt_nr" cols="60" rows="5"><?php
echo $myrow[kt_nr];?></textarea></td>
  </tr>
    <!-- ----------------------------------------------考题答案显示------------------------------------------- -->
  <tr>
    <td height="46" align="center" bgcolor="#DDDDDD">考题答案</td>
    <td colspan="3" align="center" bgcolor="#FFFFFF"><textarea name="kt_daan" cols="60" rows="5"><?php
echo $myrow[kt_daan];?></textarea></td>
  </tr>
    <!-- ----------------------------------------------考题正确答案显示------------------------------------------- -->
  <tr>
    <td height="33" align="center" bgcolor="#DDDDDD">考题正确答案</td>
    <td colspan="3" align="center" bgcolor="#FFFFFF"><textarea name="kt_zqdaan" cols="60" rows="5"> <?php
echo $myrow[kt_zqdaan];?></textarea></td>
  </tr>
</form>
<?php
 }                                                    //while 循环语句结束
?>
</table>
```

代码贴士

❶ mssql_query()函数：向 SQL Server 服务器中的当前活动数据库发送一条 SQL 查询。

❷ mssql_fetch_array()函数：返回根据从结果集获取的行生成的数组，如果没有更多行则返回 false。语法如下：

```
array mssql_fetch_array( resource result [, int result_type] )
```

☑ 参数 result：资源类型的参数，要传入的是由 mssql_query()函数返回的“数据指针”。

☑ 参数 result_type：可选项，整数型参数，要传入的是 MSSQL_ASSOC、MSSQL_NUM 和 MSSQL_BOTH 3 种由 PHP 定义好的常数之一，默认值是 MSSQL_BOTH。

- 用 MSSQL_ASSOC 只得到关联索引。
- 用 MSSQL_NUM 只得到数字索引。
- 用 MSSQL_BOTH 将得到一个同时包含关联和数字索引的数组。

查询考题信息页面提供了修改考题信息的功能，管理员可对指定的考题信息进行编辑，单击“修改”按钮后，提交 Submit2 表单信息到数据处理页 ktxx_gl_ok.php。

例程16　代码位置：光盘\TM\04\online_ks\admin\ktxx_gl_ok.php

```
<?php
include("../conn/conn.php");                          //连接数据库文件
if($Submit2==true){                                   //如果单击了“修改”按钮，则执行下面语句
/* ********************************应用 update 语句修改考题信息******************************** */
$querys=mssql_query("update tb_kt set
kt_lb='$kt_lb',kt_lx='$kt_lx',kt_fs='$kt_fs',kt_nr='$kt_nr',kt_daan='$kt_daan',kt_zqdaan='$kt_zqdaan' where kt_id='$kt_id'");
/* *********************如果考题信息修改成功，弹出提示，并重定向到首页********************** */
if($querys){
echo "<script>alert('考题更新成功！'); window.location.href='index.php?htgl=考题信息管理';</script>";
}
}
```

查询考题信息页面提供了删除考题信息的功能，管理员可对指定的考题信息进行删除，单击“删除”按钮后，提交 Submit3 表单信息到数据处理页 ktxx_gl_ok.php。

例程17　代码位置：光盘\TM\04\online_ks\admin\ktxx_gl_ok.php

```
<?php
include("../conn/conn.php");                          //连接数据库文件
if($Submit3==true){                                   //如果单击了“删除”按钮，则执行下面语句
/* ******************************应用 delete 语句删除指定的考题信息****************************** */
$query=mssql_query("delete from tb_kt where kt_id='$kt_id'");
/* *********************如果考题信息删除成功，弹出提示，并重定向到首页********************** */
if($query){
echo "<script>alert('考题类别删除成功！'); window.location.href='index.php?htgl=考题信息管理';</script>";
}
}
?>
```

4.10　开发技巧与难点分析

4.10.1　考生登录编号的获取

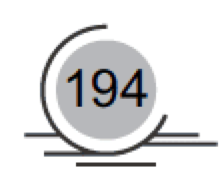
考生登录编号的生成主要应用的是 mt_rand()函数和 substr()函数。首先通过 mt_rand()函数来获取一个 100000～999999 之间的随机数，然后应用 substr()函数从该随机数中获取 6 个数字，作为考生编号。

mt_rand()函数的功能是获取随机数的值。语法格式如下：

```
int mt_rand([int min,int max])
```



说明　如果 mt_rand()函数没有提供可选参数 min 和 max，则返回 0 到 RAND_MAX 之间的伪随机数。





下面应用 mt_rand()函数来获取 100000～999999 之间的随机数。代码如下：

```
<?php echo mt_rand(100000,999999); ?>
```

运行结果为：321864

4.10.2　通过 Ajax 技术实现计时与显示剩余时间

通过 Ajax 技术实现计时与显示剩余时间，首先要创建一个 XMLHttpRequest 对象实例，确保其能够在所有支持 XMLHttpRequest 的浏览器中运行，将其代码保存在一个名称为 xmlHttpRequest.js 的文件中，然后在需要应用 Ajax 技术的页面中，应用以下代码包含该文件。

```
<script type="text/javascript" src="js/xmlHttpRequest.js"></script>
```

xmlHttpRequest.js 文件的代码如下：

例程18　代码位置：光盘\TM\04\online_ks\js\xmlHttpRequest.js

```
var xmlHttp = false;                                      //创建一个布尔型变量，用来检测是否为合法的 IE 实例
try {                                                     //检测是否使用的是 IE
      xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");  //如果 JavaScript 的版本大于 5
} catch (e) {                                             //如果不是，则使用老版本的 ActiveX 对象
      try {                                               //如果使用的是 IE 浏览器
            xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
      } catch (e2) {}
}
/* ******************如果使用的是非 IE 浏览器，则创建一个该对象的 JavaScript 实例******************** */
if (!xmlHttp && typeof XMLHttpRequest != "undefined") {
      try{
            xmlHttp = new XMLHttpRequest();
      }catch(e3){ xmlHttp = false;}
}
```

接下来编写两个自定义的 JavaScript 函数 ShowTime()和 sparetime()，通过 ShowTime()函数读取显示时间文件 ShowTime.php 中的数据；通过 sparetime()函数读取获取剩余时间文件 sparetime.php 中的数据。ShowTime()和 sparetime()函数的代码如下：

例程19　代码位置：光盘\TM\04\online_ks\js\xmlHttpRequest.js

```
<script type="text/javascript">
timer = window.setInterval("ShowTime()",1000);                //每隔一秒钟调用一次 ShowTime()函数
//定义 ShowTime()函数以通过 XMLHttpRequest 对象读取 ShowTime.php 文件中的数据
function ShowTime(){
      xmlHttp.open("post","showtime.php", true);              //以 post 方式发送一个新请求
      xmlHttp.onreadystatechange = function(){
      if(xmlHttp.readyState == 4){                            //如果是服务器响应发出的请求，则执行以下操作
            tet = xmlHttp.responseText;                       //获取返回的响应信息
            /* ********************将获取到的信息赋予指定的 DIV 标记******************** */
            document.getElementById("show_time").innerHTML = tet;
      }
}
```

```
xmlHttp.send(null);                                    //发送请求
}
</script>
<script type="text/javascript">
time = window.setInterval("sparetime()",1000);         //每隔一秒钟调用一次 sparetime()函数
/* ********定义 sparetime()函数以通过 XMLHttpRequest 对象读取 sparetime.php 文件中的数据************* */
function sparetime(){
    xmlHttp.open("post","sparetime.php", true);        //以 post 方式发送一个新请求
    xmlHttp.onreadystatechange = function(){
        if(xmlHttp.readyState == 4){                   //如果是服务器响应发出的请求，则执行以下操作
            tet = xmlHttp.responseText;                //获取返回的响应信息
            /* **********************将获取到的信息赋予指定的 DIV 标记*********************** */
            document.getElementById("sparetime").innerHTML = tet;
            if(tet=="00:00"){                          //判断当剩余时间为 00:00 时
                form1.submit();                        //提交 form1 表单中的数据
            }
        }
    }
    xmlHttp.send(null);                                //发送请求
}
</script>
```

在 ShowTime.php 文件中实现当前时间的显示。实现的原理是：获取一个在 session 变量中存储的考试开始时间的时间戳，然后再应用 mktime()函数获取当前时间的时间戳，应用当前时间戳减去考试开始时间的时间戳，最后应用 date()函数输出获取的新时间戳的时间值。代码如下：

例程20　代码位置：光盘\TM\04\online_ks\ShowTime.php

```
<?php session_start();                      //初始化 session 变量
    header('Content-Type: text/xml');
    $dates=$_SESSION[dates];                //获取 session 变量中存储的考试开始时间的时间戳
    $dates2=mktime();                       //获取系统当前时间的时间戳
    $dates3=$dates2-$dates;                 //应用当前时间的时间戳减去考试开始时间的时间戳
    echo date("i:s",$dates3);               //输出得出的时间
?>
```

在 sparetime.php 文件中获取考试的剩余时间。实现的原理是：首先设置考试时间为 20 分钟，在考试开始时间的基础上加上 20 分钟，然后用考试时间减去系统的当前时间，获取的就是考试的剩余时间，当剩余时间为 00:00 时表示本次考试结束，将考题自动提交。sparetime.php 文件的代码如下：

例程21　代码位置：光盘\TM\04\online_ks\sparetime.php

```
<?php session_start();                      //初始化 session 变量
    header('Content-Type: text/xml');
    $dates=$_SESSION[dates];                //获取 session 变量中存储的考试开始时间的时间戳
    $dates1=$dates+1*60;                    //设定考试时间
    $dates2=mktime();                       //获取系统的当前时间戳
    $dates3=$dates1-$dates2;                //计算考试的剩余时间
    echo date("i:s",$dates3);               //输出考试的剩余时间
?>
```



最后在考试操作页面中通过 DIV 标签来输出考试时间和剩余时间。代码如下：

```
<div id="sparetime"></div>                    <!--输出考试剩余时间-->
<div id="show_time"></div>                    <!--输出考试时间-->
```

4.11　Ajax 无刷新技术专题

4.11.1　Ajax 概述

Ajax 是 Asynchronous JavaScript and XML 的缩写，意思是异步的 JavaScript 与 XML。

Ajax 并不是一种新技术，或者说它不是一种技术，实际上，它是将 JavaScript、XHTML 和 CSS、DOM、XML 和 XSTL、XMLHttpRequest 等编程技术以新的强大方式组合而成，可以让开发人员构建基于 PHP 技术的 Web 应用，并打破了使用页面重载的惯例。Ajax 包含：

☑　XHTML 和 CSS 技术实现标准页面。
☑　Document Object Model 技术实现动态显示和交互。
☑　XML 和 XSLT 技术实现数据的交换和维护。
☑　XMLHttpRequest 技术实现异步数据接收。
☑　JavaScript 绑定和处理所有数据。

Ajax 是一种运用浏览器的技术，它可以在浏览器和服务器之间得到异步通信机制进行数据通信，从而允许浏览器向服务器获取少量信息而不是刷新整个页面。

4.11.2　Ajax 的优点

Ajax 是使用客户端脚本与 Web 服务器交换数据的 Web 应用开发方法。这样，Web 页面不用打断交互流程进行重新加载，就可以动态地更新。Ajax 的优点介绍如下：

（1）减轻服务器的负担。

Ajax 的原则是“按需取数据”，可以最大程度地减少冗余请求，从而减轻对服务器造成的负担。

（2）无刷新更新页面，减少用户心理和实际的等待时间。

“按需取数据”的模式减少了数据的实际读取量。如果说重载的方式是从一个终点回到原点再到另一个终点的话（如图 4.28 所示），那么 Ajax 就是以一个终点为基点到达另一个终点（如图 4.29 所示）。

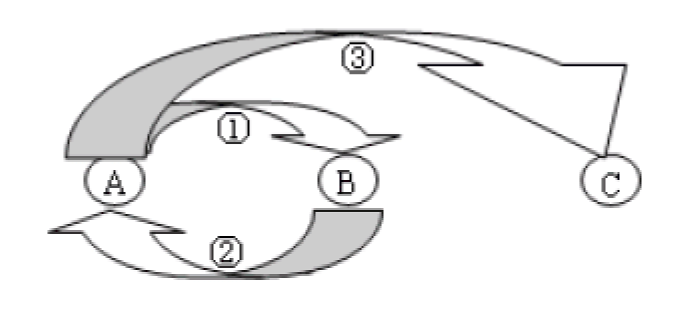


图 4.28　重载方式

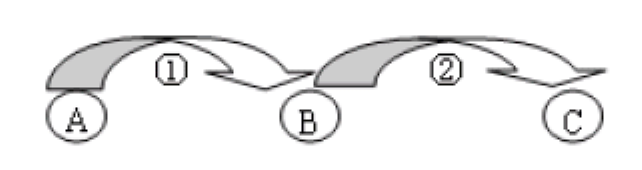


图 4.29　Ajax 方式

其次，即使要读取较大的数据，也不会出现白屏的情况。Ajax 使用 XMLHTTP 对象发送请求并得到服务器响应，在不重新载入整个页面的情况下用 JavaScript 操作 DOM 最终更新页面，所以在读取数据的过程中，用户所面对的不是白屏，而是原来的页面状态；页面只有接收到全部数据后才更新相应部分的内容，而这种更新也是瞬间的，用户几乎感觉不到。

（3）带来更好的用户体验。

（4）把部分服务器负担的工作转交给客户端，利用客户端闲置的能力来处理任务，从而减轻服务器和带宽的负担，节约空间和宽带租用成本。

（5）可以调用外部数据。

（6）是一种基于标准化并被广泛支持的技术，不需要下载插件或者小程序。

（7）进一步促进 Web 页面展现形式与数据的分离。

4.11.3 Ajax 的工作原理

传统的 Web 模式强制用户进入“提交→等待→重新显示”网页，用户的动作总是与服务器进行同步思考，客户在网页上的操作转化为 Http 请求传回服务器，而服务器接受请求以及相关数据，然后解析数据并将其发送给相应的处理单元后，将返回的数据转成 HTML 页返还给客户。而当服务器处理数据时，用户只能等待，每一步操作都需要等待服务器返回新的网页。由于每次应用的交互都需要向服务器发送请求，应用的响应时间就依赖于服务器的响应时间，这就导致了用户页面的响应比本地应用慢得多。

运用了 Ajax 技术的 Web 应用模型，它的工作原理相当于在客户端和服务器端之间添加了一个中间层，称为 Ajax 引擎（采用 JavaScript 编写，通常在一个隐藏的框架中），实现了与服务器进行异步思考的通信能力，从而使用户从请求/响应的循环中解脱出来，向服务器发出异步请求，也就是不用等待服务器的通信。所以用户不用再打开一个空白窗口，等待服务器完成后再进行响应。Ajax 应用可以仅向服务器发送并取回必需的数据，它使用 SOAP 或其他一些基于 XML 的 Web Service 接口，并在客户端采用 JavaScript 处理来自服务器的响应。因为在服务器和浏览器之间交换的数据大量减少，所以 Web 站点看起来是即时响应的。同时很多的处理工作可以在发出请求的客户端机器上完成，所以 Web 服务器的处理时间也减少了。

引入 Ajax 的 Web 模型与传统的 Web 模型比较如图 4.30 所示。

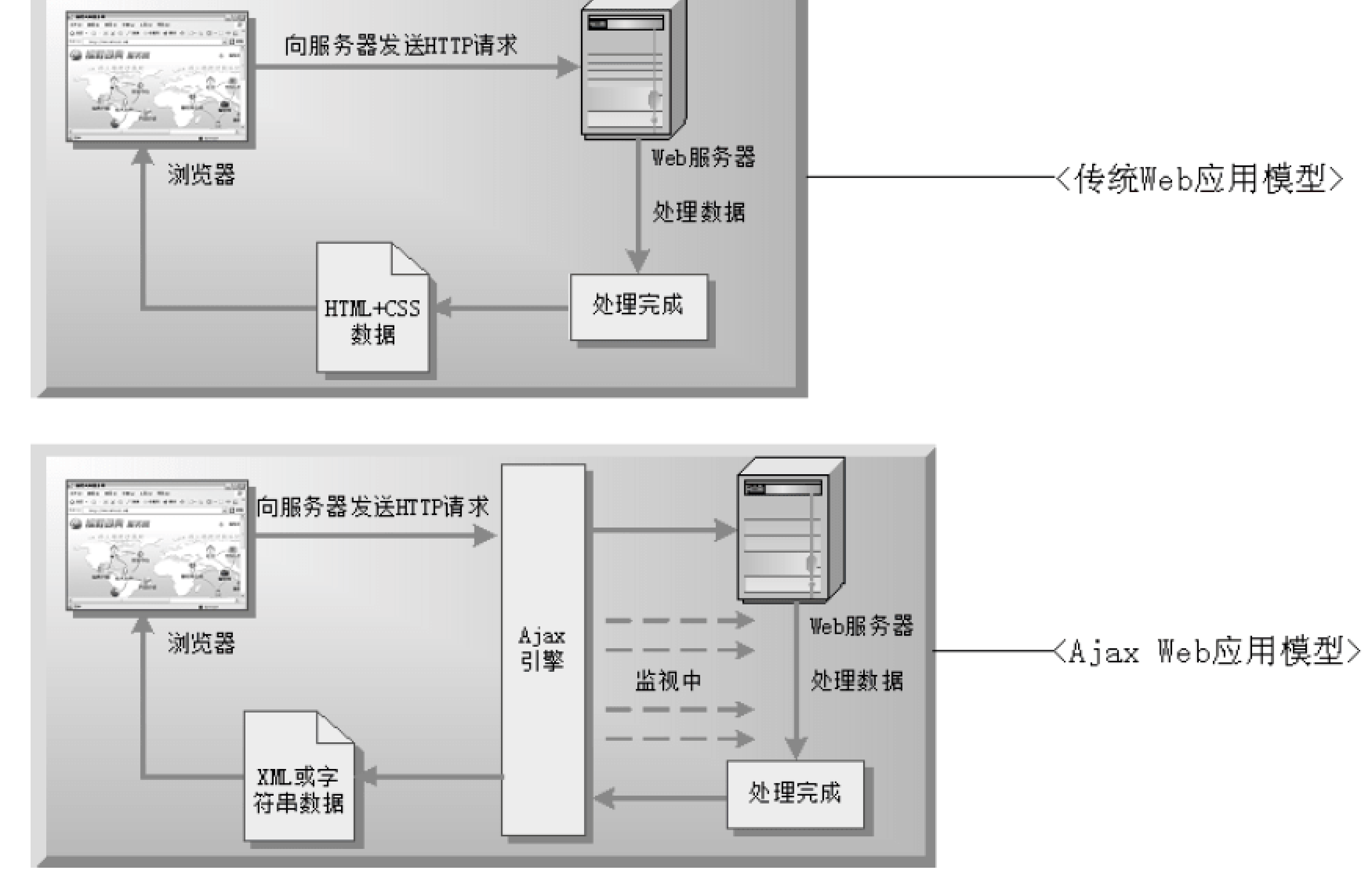


图 4.30　传统 Web 应用模型与 Ajax Web 应用模型的比较



4.11.4　Ajax 的工作流程

使用 Ajax，用户可以创建接近本地桌面应用的更直接、更可用、更丰富、更动态的 Web 用户界面。Ajax 内部的工作流程如图 4.31 所示。

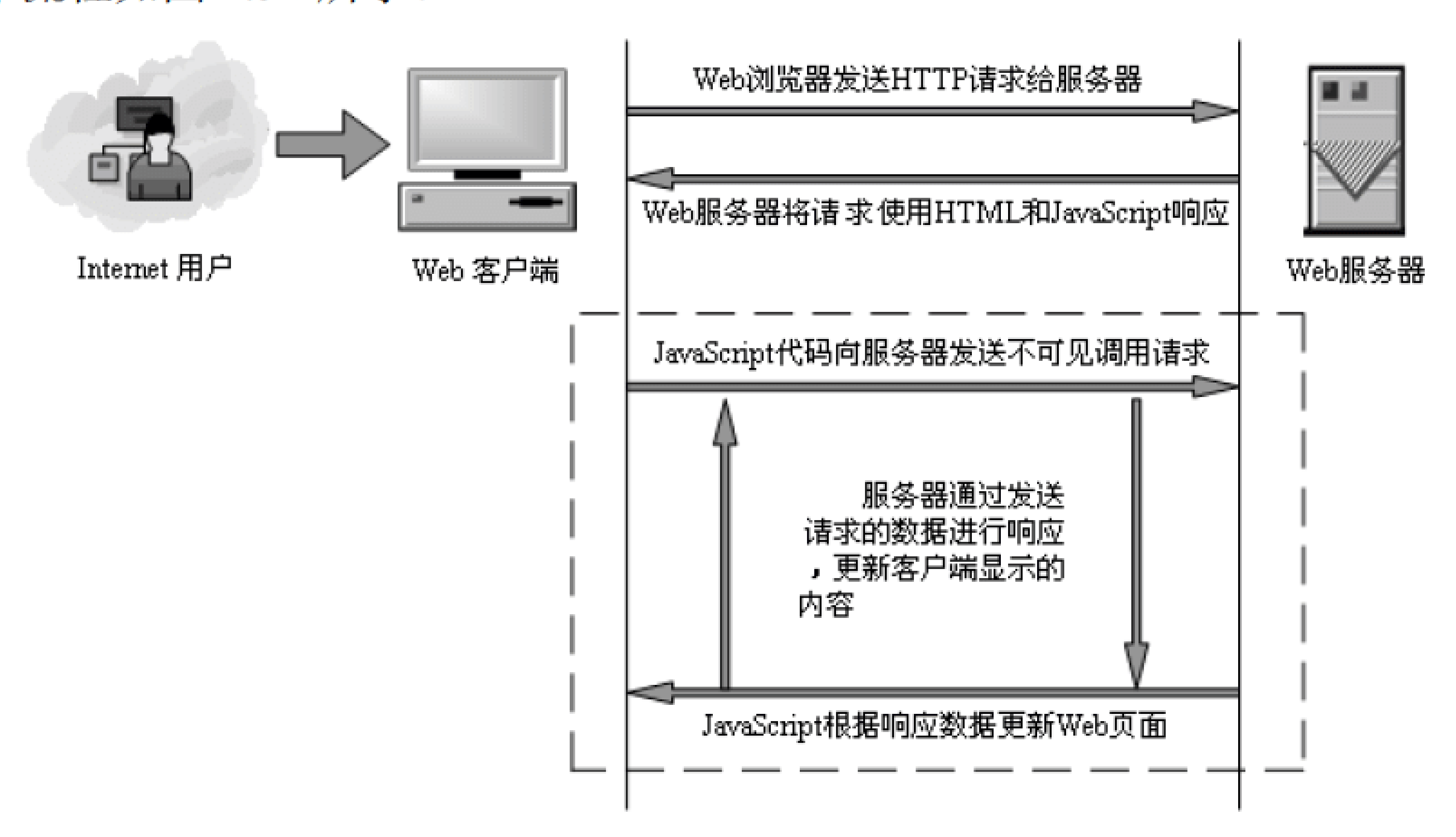


图 4.31　Ajax 内部工作流程图

4.11.5　Ajax 中的核心技术 XMLHttpRequest

Ajax 技术中最核心的技术就是 XMLHttpRequest，它是一个具有应用程序接口的 JavaScript 对象，能够使用超文本传输协议（HTTP）连接一个服务器，是微软公司为了满足开发者的需要，于 1999 年在 IE 5.0 浏览器中率先推出的。现在许多浏览器都对其提供了支持，不过实现方式与 IE 有所不同。

通过 XMLHttpRequest 对象，Ajax 可以像桌面应用程序一样只同服务器进行数据层面的交换，而不用每次都刷新页面，也不用每次都将数据处理的工作交给服务器来做，这样既减轻了服务器负担，又加快了响应速度，而且还缩短了用户等待的时间。

在使用 XMLHttpRequest 对象发送请求和处理响应之前，首先需要初始化该对象，由于 XMLHttpRequest 不是一个 W3C 标准，所以对于不同的浏览器，初始化的方法也是不同的。

1．IE 浏览器

IE 浏览器把 XMLHttpRequest 实例化为一个 ActiveX 对象。具体方法如下：

```
var http_request = new ActiveXObject("Msxml2.XMLHTTP");
```

或者

```
var http_request = new ActiveXObject("Microsoft.XMLHTTP");
```

说明　上面语法中的 Msxml2.XMLHTTP 和 Microsoft.XMLHTTP 是针对 IE 浏览器的不同版本而进行设置的，目前比较常用的是这两种。

2. Mozilla、Safari 等其他浏览器

Mozilla、Safari 等其他浏览器把它实例化为一个本地 JavaScript 对象。具体方法如下：

```
var http_request = new XMLHttpRequest();
```

为了提高程序的兼容性，可以创建一个跨浏览器的 XMLHttpRequest 对象。创建一个跨浏览器的 XMLHttpRequest 对象其实很简单，只需要判断一下不同浏览器的实现方式，如果浏览器提供了 XMLHttpRequest 类，则直接创建一个实例，否则使用 IE 的 ActiveX 控件。具体代码如下：

```
if (window.XMLHttpRequest) { // Mozilla、Safari…
    http_request = new XMLHttpRequest();
} else if (window.ActiveXObject) { // IE 浏览器
    try {
        http_request = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            http_request = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (e) {}
    }
}
```

说明 由于 JavaScript 具有动态类型特性，而且 XMLHttpRequest 对象在不同浏览器上的实例是兼容的，所以用同样的方式访问 XMLHttpRequest 实例的方法即可，不需要再考虑创建该实例的方法是什么。

4.12 本章总结

至此，一个完整的网络在线考试系统已经全部完成。在程序的开发过程中，采用了 switch 框架，使整个系统的设计思路更加清晰。同时，为了使程序更加人性化，系统中应用了 Ajax 技术实现在线考试时自动计时等功能。希望读者能认真学习，并做到融会贯通。

第 5 章

物流配送信息网

（Apache+PHP+phpMyAdmin+MySQL 5.0 实现）

物流信息化是指物流企业运用现代信息技术对物流过程中产生的全部或部分信息进行采集、分类、传递、汇总、查询等一系列处理活动，以实现对货物流动过程的控制，从而降低成本、提高效益的管理活动。

物流企业信息化的目的就是要满足企业自身管理的需要和不同类型企业在物流业务外包过程中对信息交换方的要求，也就是通过建设物流信息系统，提高信息流转效率，降低物流运作成本。

目前我国正处于全面推进信息化的进程之中，所以物流领域的信息化已成为一个必然。物流信息化将成为现代物流的灵魂，将是现代物流发展的必经之路。

本章开发的物流配送信息网就是为了增强企业的市场竞争力，加快企业的信息化进程，使企业在市场竞争中立于不败之地。通过阅读本章，可以学习到：

- 物流配送系统开发的基本过程
- 如何创建 MySQL 数据库的存储过程
- 如何通过 MySQL 数据库存储过程实现用户登录
- 如何实现数据库中数据的模糊查询技术
- 如何创建浮动框架
- 如何实现订单的打印技术

5.1 开 发 背 景

随着我国信息化进程的全面推进，各领域的信息化进程都在飞速的发展，同样也推动着物流领域的信息化进程飞快的向前发展。由于信息化进程的全面推进，对现代物流提出更高的要求：信息化、自动化、网络化、智能化和柔性化等。物流行业的竞争日益激烈，客户需求的标准也越来越高，物流企业要想在市场中占有一席之地，必须要建立一个高效、快捷、方便的物流信息系统，为客户提供一流的服务，并且能够及时、准确地掌握客户的需求，对客户的需求做出快速的反映，在最短的时间内以最大限度挖掘和优化物流资源来满足客户的需求，从而建立高效的数字化物流经济。

5.2 系 统 分 析

5.2.1 需求分析

随着现代物流信息化进程的加快，传统的物流管理方式已经不再适应当前物流发展的要求，取而代之的将是以计算机为基础的网络化物流管理方式。某物流配送公司为适应物流信息化进程的发展，急需开发一个物流配送系统，通过网络来实现对物流操作流程进行管理，不但可以为企业的运营过程节省大量的人力、物力、财力和时间，提高物流系统运行的效率，而且可以使企业在客户心中树立了一个全新的形象，为企业的发展奠定一个良好的基础。现根据对该物流配送公司的实际调查，以及公司的具体要求，制定出物流配送信息网的规划方案。具体内容如下：

- ☑ 网站页面设计要求美观大方，能够展示企业形象。
- ☑ 为企业在客户中树立一个全新的形象。
- ☑ 网站的操作流程简单、方便，能够提高工作效率。
- ☑ 提供物流配送的全程跟踪。
- ☑ 提供配送信息的及时查询。
- ☑ 实现对配送车辆的管理。
- ☑ 实现对客户信息的管理。
- ☑ 实现发货单打印的功能。

5.2.2 可行性分析

物流配送信息网的开发不但能使物流企业走上科学化、网络化管理的道路，而且能够为企业带来巨大的经济效益和技术上飞速的发展。

1．经济性

科学的管理方法，便捷的操作环境，系统的经营模式，将为企业带来更多的客户资源，树立企业的品牌形象，提高企业的经济效益。

2．技术性

网络化的物流管理方式，在操作过程中能够快捷地查找出车源信息、客户订单以及客户信息；能



够对货物进行全程跟踪，了解货物的托运情况，从而使企业能够根据实际情况，做好运营过程中的各项准备工作，作出及时准确的调整；能够保证托运人以及收货人对货物进行及时的处理。

5.3　系 统 设 计

5.3.1　系统目标

结合目前网络上物流配送系统的设计方案，对客户做的调查结果以及企业的实际需求，本项目在设计时应该满足以下目标：

☑　界面设计美观大方、操作简单。
☑　功能完善、结构清晰。
☑　能够快速查询车源信息。
☑　能够准确填写订单。
☑　能够实现订单查询、打印。
☑　能够实现对回单处理。
☑　能够对车源信息进行添加、修改和删除。
☑　能够对客户信息进行管理。
☑　能够及时、准确地对网站进行维护和更新。
☑　良好的数据库系统支持。
☑　系统运行稳定，具备良好的防范措施。

5.3.2　系统功能结构

结合需求分析和系统目标中的内容，物流配送信息网的功能结构已经设计完成。为了使读者能够更清楚地了解网站的结构，下面给出物流配送信息网的功能模块结构图和工作流程图。

物流配送信息网的功能模块结构图如图 5.1 所示。

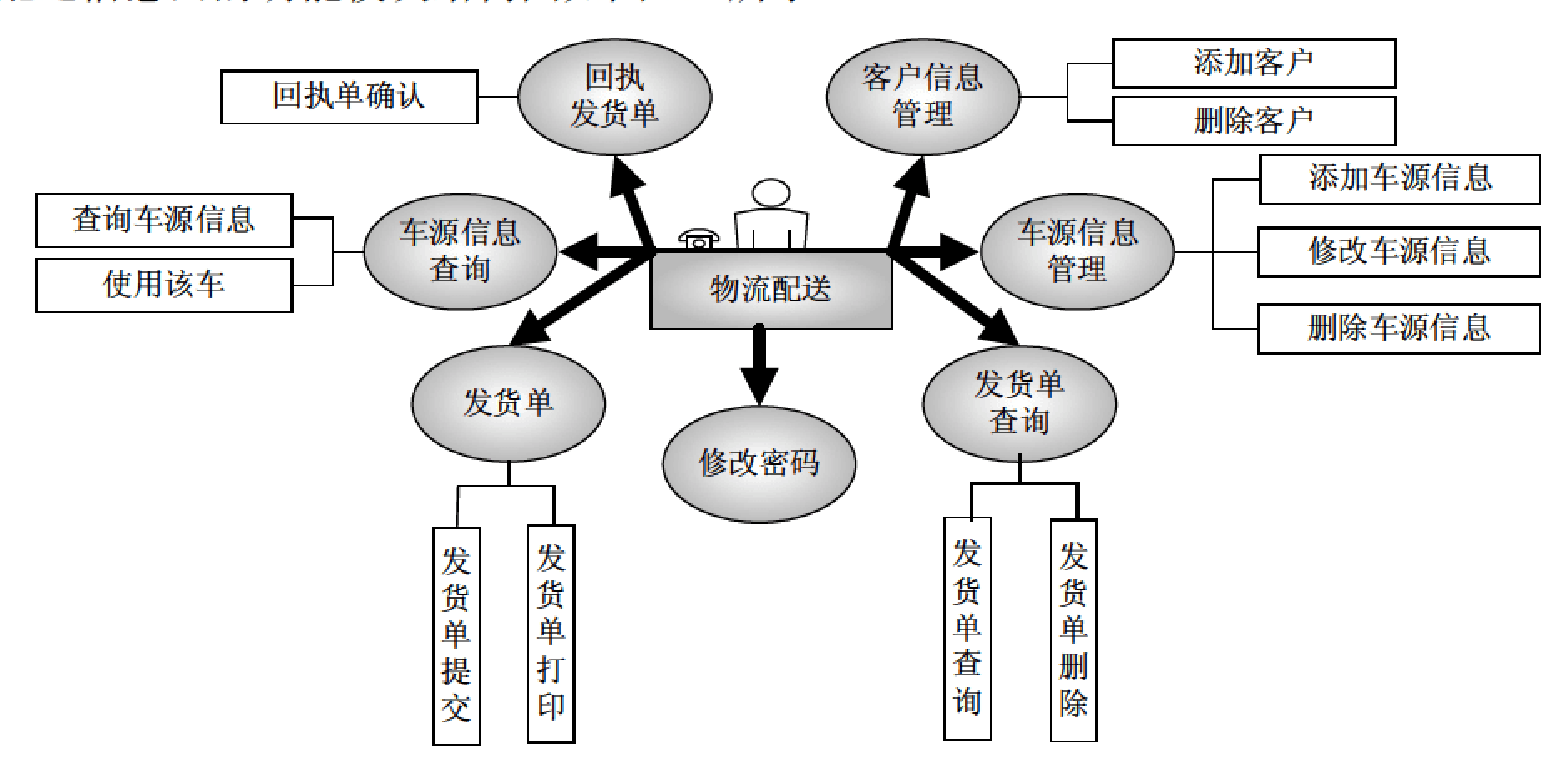


图 5.1　物流配送信息网功能模块结构图

物流配送信息网的工作流程图如图 5.2 所示。

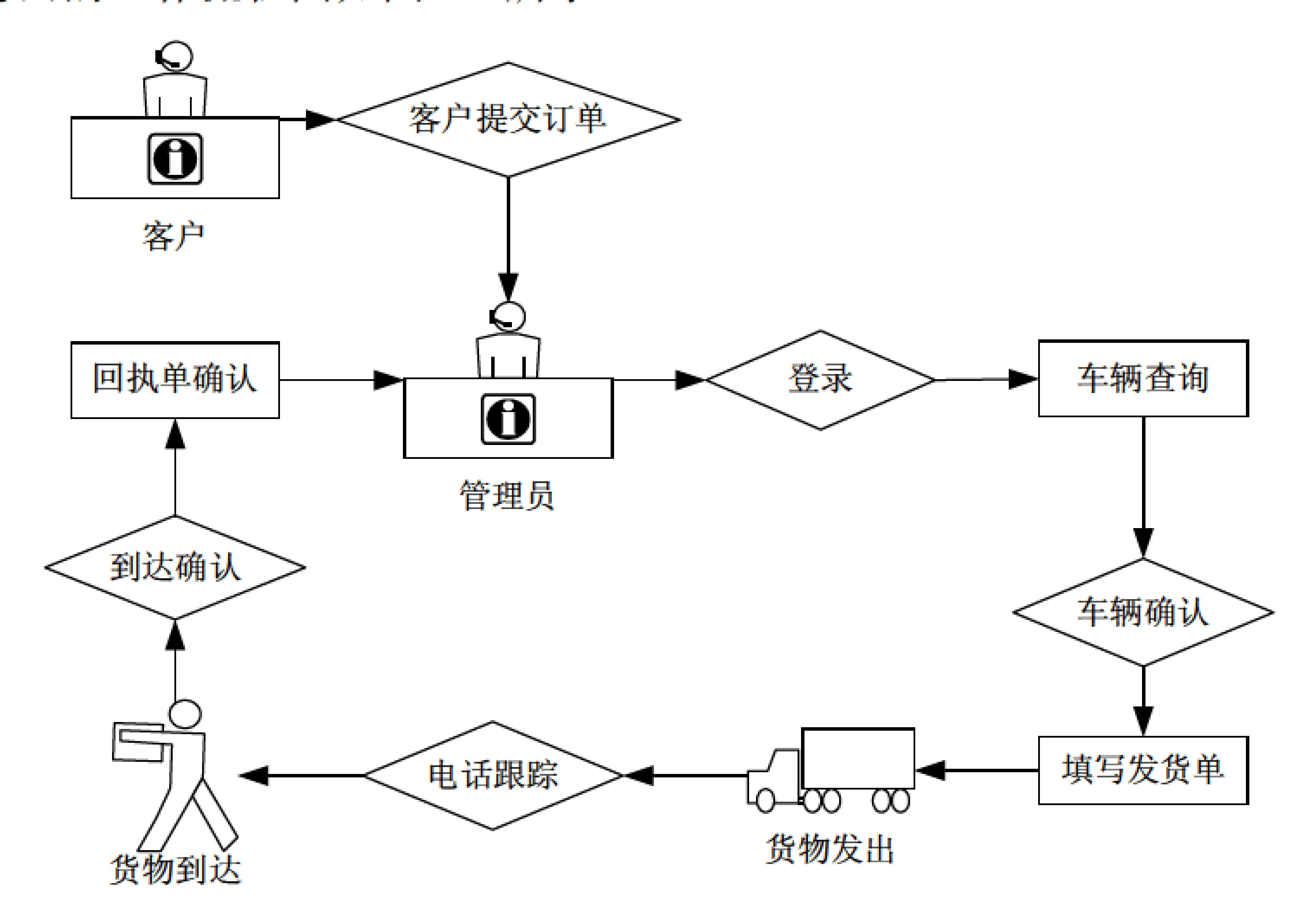


图 5.2　物流配送信息网的工作流程图

5.3.3　系统预览

物流配送信息网由多个程序页面组成，下面给出几个典型页面，其他页面参见光盘中的源程序。物流配送信息网主页如图 5.3 所示，该页面用于展示本系统的车源信息查询模块。登录页面如图 5.4 所示。

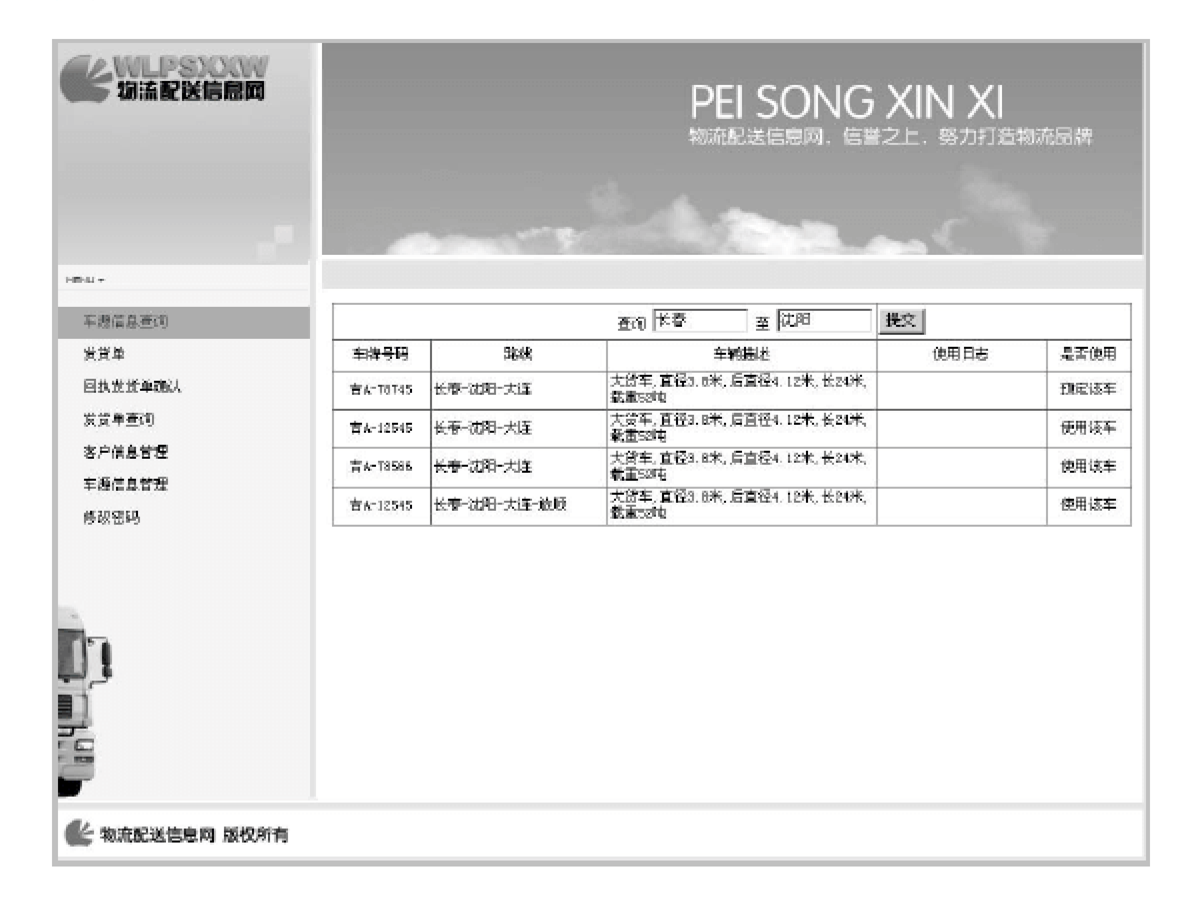



图 5.3　主页（光盘\TM\05\wlgl\indexs.php）



图 5.4　登录页面（光盘\TM\05\wlgl\index.php）

回执发货单确认模块的页面效果如图 5.5 所示，该页面主要用于实现对回执发货单的处理。发货单查询模块的页面效果如图 5.6 所示，该页面主要用于实现对指定发货单的查询，并且输出发货单的详细内容。

车源信息管理模块的页面效果如图 5.7 所示，该页面实现对车源信息的添加、修改和删除操作。修改管理员密码模块的页面效果如图 5.8 所示，该页面实现对管理员密码的修改。







图 5.5　回执发货单（光盘\TM\05\wlgl\select_dhd.php）

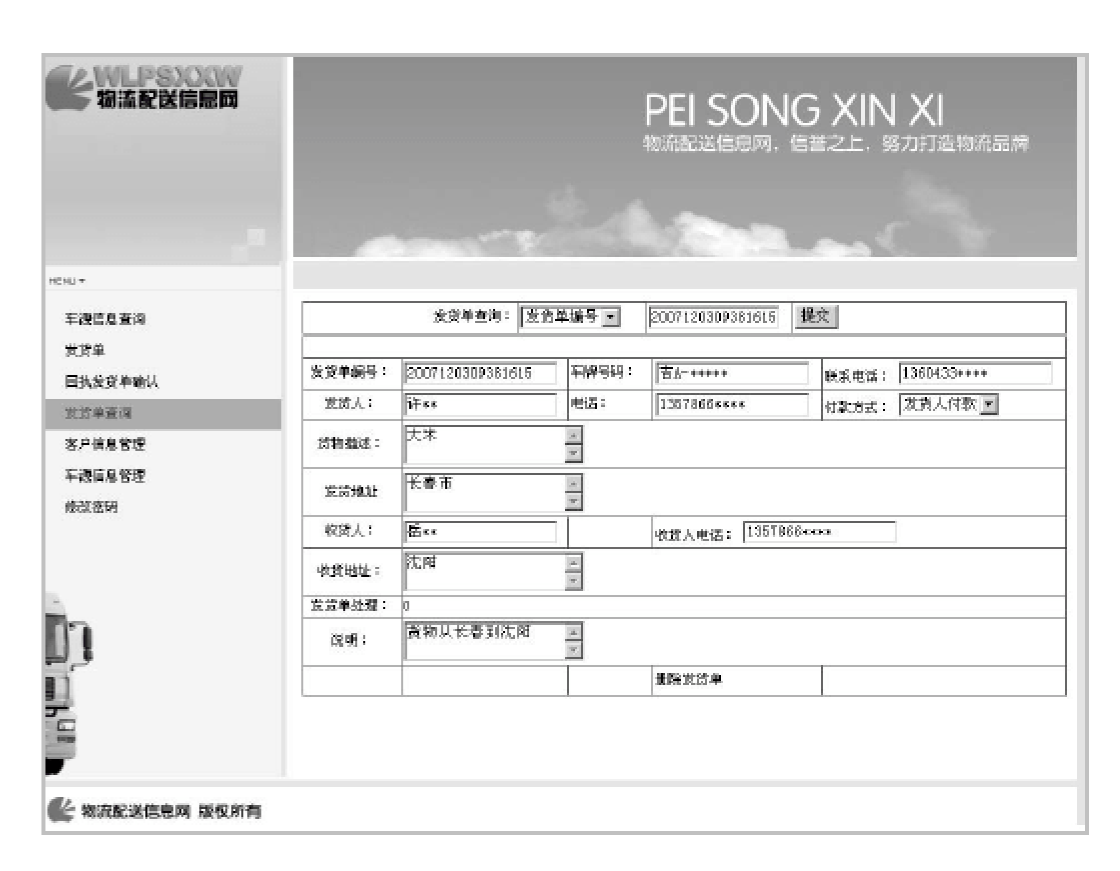


图 5.6　发货单查询（光盘\TM\05\wlgl\ hwys.php）



图 5.7　车源信息管理（光盘\TM\05\wlgl\car.php）



图 5.8　修改管理员密码（光盘\TM\05\ wlgl\update_pass.php）

5.3.4　开发环境

在开发物流配送信息网时，该项目使用的软件开发环境如下。

1．服务器端

☑　操作系统：Windows Server 2003/Linux（推荐）。
☑　服务器：Apache 2.2.8。
☑　PHP 软件：PHP 5.2.6。
☑　数据库：MySQL 5.0.51。
☑　MySQL 图形化管理软件：phpMyAdmin-2.10.3。
☑　开发工具：Dreamweaver 8。
☑　浏览器：IE 6.0 及以上版本。
☑　分辨率：最佳效果为 1024×768 像素。

2．客户端

☑　浏览器：IE 6.0 及以上版本。

☑ 分辨率：最佳效果为 1024×768 像素。

5.3.5 文件夹组织结构

在进行网站开发之前，要对网站的整体文件夹组织架构进行规划。对网站中使用的文件进行合理的分类，分别放置于不同的文件夹下。通过对文件夹组织架构的规划，可以确保网站文件目录明确、条理清晰，同样也便于网站后期的更新和维护。物流配送信息网的文件夹组织结构如图 5.9 所示。

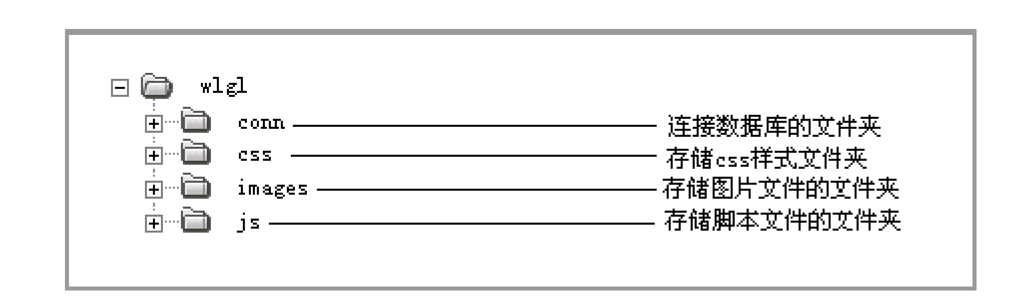


图 5.9 文件夹组织结构

5.4 数据库设计

物流配送信息网必须拥有数据库的支持，所有物流配送的数据都应该存储到数据库中，便于管理员查找车辆、订单和客户的信息。如果没有数据库的支持，那么物流配送信息网将没有任何意义。本节将对物流配送信息网的数据库设计进行详细介绍。

5.4.1 数据库分析

物流配送信息网是一个中小型的企业管理系统，考虑到开发的成本、搭配的合理性以及操作的灵活性等，使用 MySQL 数据库是最佳的选择。MySQL 数据库是完全免费的，使用它不需要任何费用，可以直接从网上免费下载；MySQL 数据库的操作也非常方便，不但可以在命令模式下操作，而且配备目前比较流行的图形化管理工具 phpMyAdmin，能够轻松地实现对 MySQL 数据库的管理和操作。

5.4.2 数据库概念设计

根据上述各节对物流配送信息网做的需求分析和系统设计，整理出物流配送信息网的实体关系 E-R 图。其中包括管理员信息实体、车源信息实体、车辆日志信息实体、客户信息实体和发货单信息实体。

1. 管理员信息实体

管理员信息实体用于存储管理员的登录名称和密码信息，包括用户名和密码两项内容。管理员信息实体的 E-R 图如图 5.10 所示。

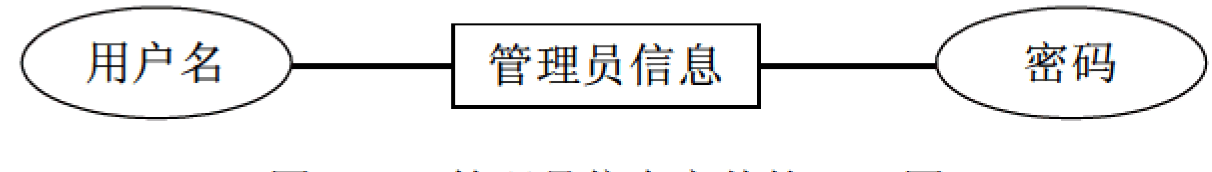


图 5.10 管理员信息实体的 E-R 图

2. 车源信息实体

车源信息实体用于存储企业拥有的车辆信息，包括车主姓名、车主身份证号、车牌号码、车主联系电话、车主家庭地址、车辆行驶路线和车辆描述。车源信息实体的 E-R 图如图 5.11 所示。





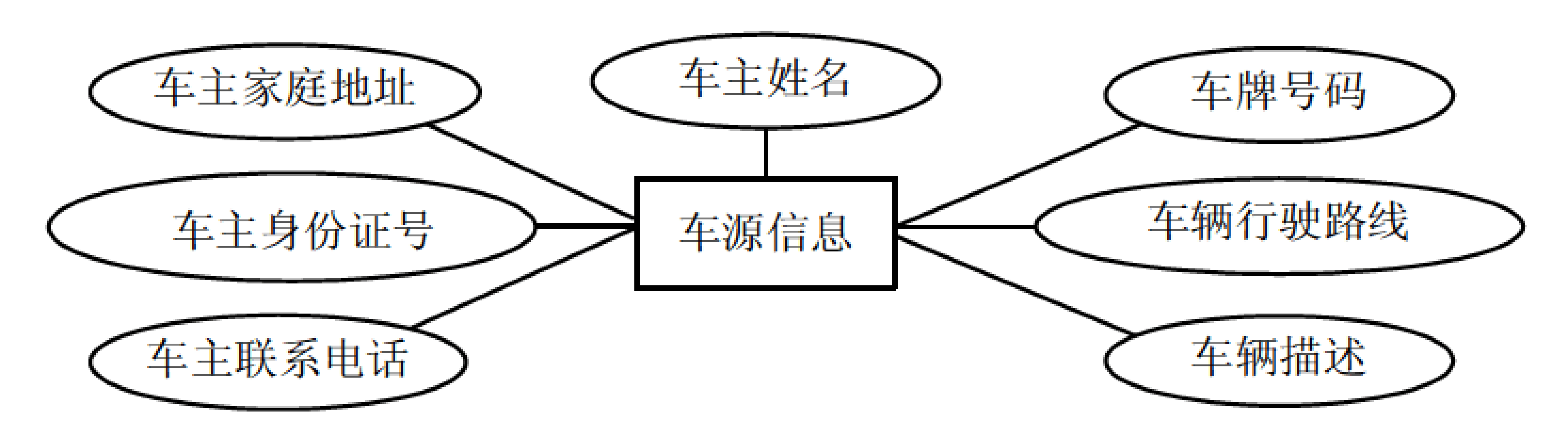


图 5.11　车源信息实体的 E-R 图

3．车辆日志信息实体

车辆日志信息实体记录车辆当前是否被占用，以及使用的时间、执行的任务，包括车牌号码、车辆日志信息（详细的车辆使用描述）、日志时间和发货单 ID（即执行的任务）。车辆日志信息实体的 E-R 图如图 5.12 所示。

4．客户信息实体

客户信息实体用于存储客户的信息，包括客户姓名、客户电话和客户联系地址。客户信息实体的 E-R 图如图 5.13 所示。

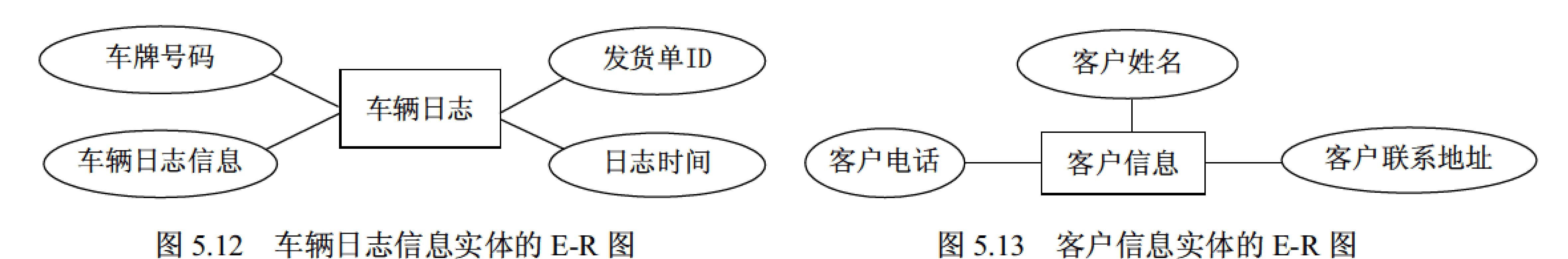


图 5.12　车辆日志信息实体的 E-R 图　　图 5.13　客户信息实体的 E-R 图

5．发货单信息实体

发货单信息实体用于存储客户填写的发货单信息，包括车牌号码、车主电话、货物描述、发货人、发货时间、发货人电话、发货地址、收货人、收货人电话、收货地址和付款方式。发货单信息实体的 E-R 图如图 5.14 所示。

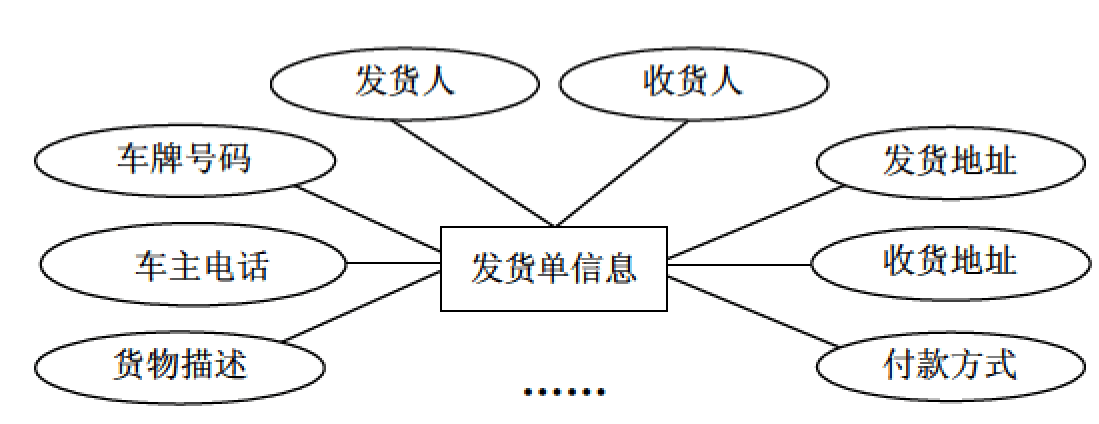


图 5.14　发货单信息实体的 E-R 图

5.4.3　创建数据库及数据表

在物流配送信息网中应用的是 db_wlgl 数据库，其中涉及 5 个数据表，如图 5.15 所示。

下面介绍物流配送信息网中使用的数据表。

1．tb_admin（管理员信息表）

管理员信息表主要用于存储管理员的登录名和密码。该数据表的结构如图 5.16 所示。

服务器: localhost ▸ 数据库: db_wlgl

表	类型	整理	说明
tb_admin	MyISAM	gb2312_chinese_ci	管理员信息表
tb_car	MyISAM	gb2312_chinese_ci	车源信息表
tb_car_log	MyISAM	gb2312_chinese_ci	车辆日志信息表
tb_customer	MyISAM	gb2312_chinese_ci	客户信息表
tb_shopping	MyISAM	gb2312_chinese_ci	发货单信息表

图 5.15　物流配送信息网的数据库

服务器: localhost ▸ 数据库: db_wlgl ▸ 表 : tb_admin

字段	类型	整理	属性	Null	默认	额外	说明
id	int(10)			否		auto_increment	管理员信息表的主键
admin_user	varchar(50)	gb2312_chinese_ci		否			登录名称
admin_pass	varchar(50)	gb2312_chinese_ci		否			登录密码

图 5.16　管理员信息表结构

2. tb_car（车源信息表）

车源信息表主要用于存储配送公司拥有的车辆信息，主要包括车主姓名、车主的身份证号码、车牌号码、车主电话、车主联系地址、车辆行驶路线和车辆描述。该数据表的结构如图 5.17 所示。

3. tb_car_log（车辆日志信息表）

车辆日志信息表主要用于存储车辆的使用情况信息，主要包括车牌号码、车辆日志信息、日志创建时间和发货单的 ID。该数据表的结构如图 5.18 所示。

服务器: localhost ▸ 数据库: db_wlgl ▸ 表 : tb_car

字段	类型	整理	属性	Null	默认	额外	说明
id	int(10)			否		auto_increment	车源信息表的主键
username	varchar(50)	gb2312_chinese_ci		否			车主的姓名
user_number	varchar(50)	gb2312_chinese_ci		否			车主的身份证号码
car_number	varchar(50)	gb2312_chinese_ci		否			车牌号码
tel	varchar(50)	gb2312_chinese_ci		否			车主电话
address	varchar(80)	gb2312_chinese_ci		否			车主联系地址
car_road	mediumtext	gb2312_chinese_ci		否			车辆行驶路线
car_content	mediumtext	gb2312_chinese_ci		否			车辆描述

图 5.17　车源信息表结构



服务器: localhost ▸ 数据库: db_wlgl ▸ 表 : tb_car_log

字段	类型	整理	属性	Null	默认	额外	说明
log_id	int(10)			否		auto_increment	车辆日志表的ID
car_number	varchar(50)	gb2312_chinese_ci		否			车牌号码
car_log	mediumtext	gb2312_chinese_ci		否			车辆日志信息
log_date	datetime			否			日志创建时间
fahuo_id	varchar(50)	gb2312_chinese_ci		否			发货单的ID

图 5.18　车辆日志信息表结构

4. tb_customer（客户信息表）

客户信息表主要用于存储客户的信息，包括客户姓名、客户电话以及联系地址。该数据表的结构如图 5.19 所示。

5. tb_shopping（发货单信息表）

发货单信息表主要用于存储客户填写的发货单中的信息，主要包括车牌号码、车主电话、货物描述、发货人、发货时间、发货人电话、发货地址、收货人、收货人电话、收货地址和付款方式。该数据表的结构如图 5.20 所示。

服务器: localhost ▸ 数据库: db_wlgl ▸ 表 : tb_customer

字段	类型	整理	属性	Null	默认	额外	说明
customer_id	int(10)			否		auto_increment	客户信息表的ID
customer_user	varchar(50)	gb2312_chinese_ci		否			客户姓名
customer_tel	varchar(50)	gb2312_chinese_ci		否			客户电话
customer_address	varchar(80)	gb2312_chinese_ci		否			联系地址

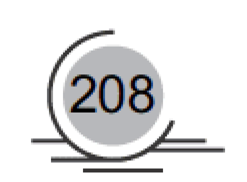
图 5.19　客户信息表结构



服务器: localhost ▸ 数据库: db_wlgl ▸ 表 : tb_shopping

字段	类型	整理	属性	Null	默认	额外	说明
id	int(10)			否		auto_increment	发货单信息表的主键ID
car_number	varchar(50)	gb2312_chinese_ci		否			车牌号码
fahuo_content	mediumtext	gb2312_chinese_ci		否			货物描述
fahuo_id	varchar(50)	gb2312_chinese_ci		否			发货ID
fahuo_user	varchar(50)	gb2312_chinese_ci		否			发货人
fahuo_time	datetime			否			发货时间
fahuo_ys	varchar(20)	gb2312_chinese_ci		是	0		回单确认
fahuo_fk	varchar(20)	gb2312_chinese_ci		否			付款方式
car_tel	varchar(50)	gb2312_chinese_ci		否			车主电话
shouhuo_user	varchar(50)	gb2312_chinese_ci		否			收货人
shouhuo_address	mediumtext	gb2312_chinese_ci		否			收货地址
fahuo_address	mediumtext	gb2312_chinese_ci		否			发货地址
fahuo_tel	varchar(50)	gb2312_chinese_ci		否			发货人电话
shouhuo_tel	varchar(50)	gb2312_chinese_ci		否			收货人电话

图 5.20　发货单信息表结构





5.5　网站首页设计

5.5.1　网站首页概述

网站首页是整个网站的门面，是浏览者看到的第一视觉界面，所以在设计网站的首页时应该将网站中主要的内容尽量展示给浏览者，让浏览者能够更快地了解网站的内容。物流配送信息网的首页主要包括如下功能。

☑　车源信息查询：主要实现车辆信息查询，及时为客户选择货物配送的方案。
☑　发货单：主要用于客户发货单信息填写。
☑　回执发货单确认：主要用于对货物配送回执单的确认。
☑　发货单查询：主要用于对发货单进行查询。
☑　客户信息管理：主要用于对客户的信息进行管理。
☑　车源信息管理：主要用于对企业拥有的车辆信息进行管理。
☑　修改密码：主要用于对管理员登录的密码进行修改。

本案例中提供的网站首页如图 5.21 所示。该页面在本书光盘中的路径为\TM\05\wlgl\index.php。

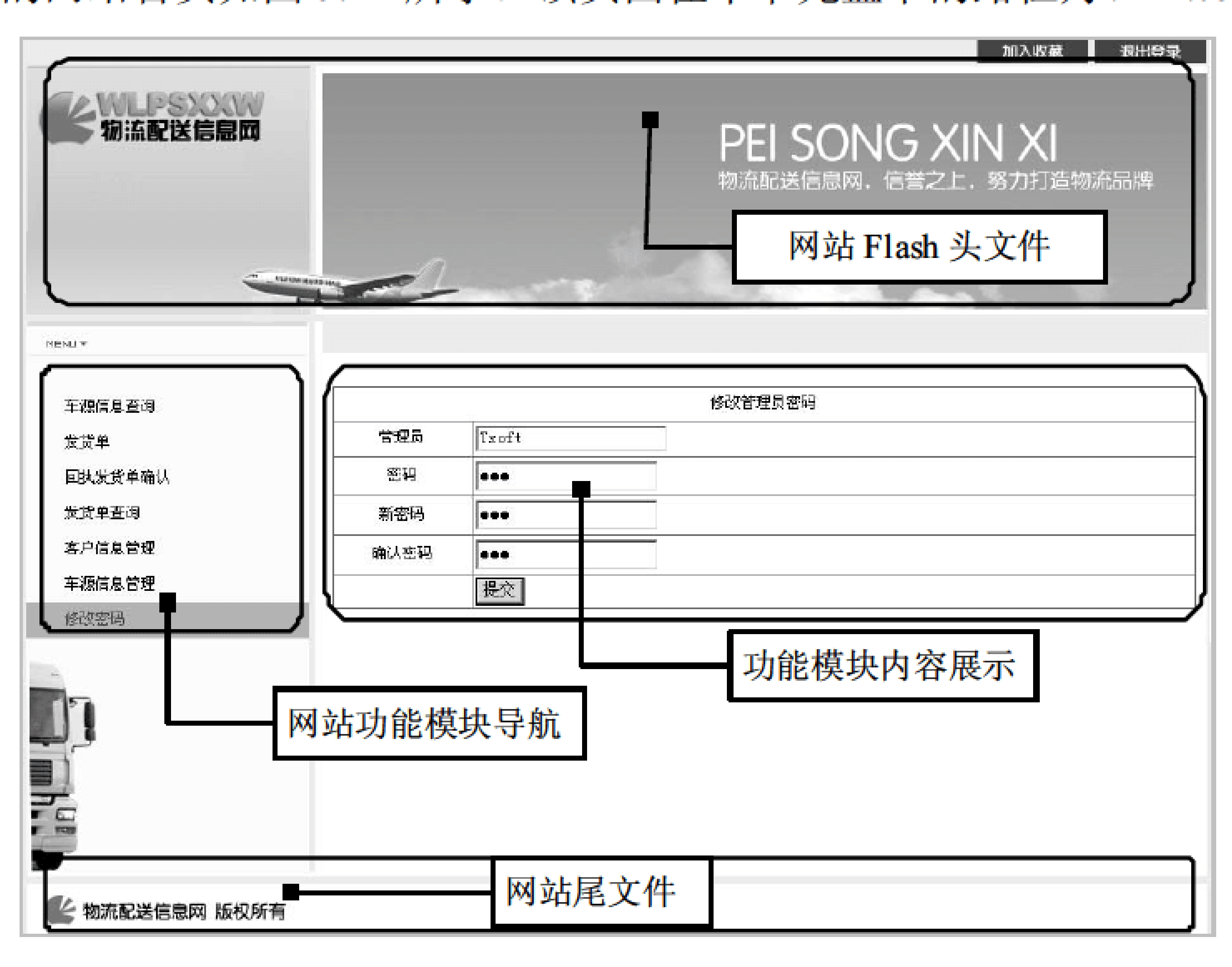


图 5.21　物流配送信息网首页

5.5.2　网站首页技术分析

物流配送信息网是为物流配送公司设计的一个物流管理系统，其页面的设计突出简洁、方便，功能的实现以便于操作和维护为根本。网站首页的页面设计如图 5.21 所示，由一个简单的框架组成，其结构由两部分组成：一部分用于输出网站中包括的模块；另一部分用于输出对应模块中的内容。首页框架结构如图 5.22 所示。

图 5.22　首页框架结构

这里使用的框架结构不是 frame 框架，而是应用 switch 语句，加上表格的嵌套实现的框架效果。有关 switch 语句的详细应用可以参考 2.12.2 节。

5.5.3　网站首页的实现过程

物流配送信息网的首页设计非常简单，主要应用一个简单的框架功能，并且以超链接作为辅助，最终实现在不同功能模块之间的跳转。

例程01　代码位置：光盘\TM\05\wlgl\indexs.php

首先连接数据库，判断登录用户的身份，如果是管理员，则可以进入网站进行操作；否则将提示“请您正确登录！”，并且跳转到网站的登录页面。关键代码如下：

```
<?php    session_start();                                   //初始化 session 变量
include("conn/conn.php");                                   //连接数据库
if($_SESSION[admin_user]==true){                            //判断登录用户的身份
?>
<!--省略了部分代码-->
<?php
}else{                                                      //如果没有以管理员登录，则跳转到登录页
    echo "<script>alert('请您正确登录!'); window.location.href='index.php';</script>";
}
?>
```

然后设计首页中使用的模块超链接，并且为每个超链接定义变量标识，当单击不同功能模块超链接时提交不同的变量标识，作为下一步获取模块中详细内容的依据；其中还使用 ononMouseOver 和 onMouseOut 事件，控制鼠标移动到和离开单元格时显示不同的颜色。部分代码如下：

```
<!-- --------------------应用 onMouseOver 和 onMouseOut 事件控制鼠标移过和离开时的时间---------------------- -->
<tr onMouseOver="this.bgColor='#9FCB3A'"onMouseOut="this.bgColor='#F5F5F5'" >
    <td height="28" align="center"> </td>
    <!-- ------------------设置车源信息查询模块的超链接，变量标识为"lmbs=车源信息查询"---------------- -->
    <td align="left"><a href="indexs.php?lmbs=车源信息查询">车源信息查询</a></td>
</tr>
<tr onMouseOver="this.bgColor='#9FCB3A'"onMouseOut="this.bgColor='#F5F5F5'" >
    <td height="28" align="center"> </td>
    <!-- -------------------------设置发货单模块的超链接，变量标识为"lmbs=发货单"------------------------- -->
    <td align="left"><a href="indexs.php?lmbs=发货单">发货单</a></td>
</tr>
```

最后应用 switch 语句，根据变量标识$lmbs 提交的值进行判断，使用 include 包含语句调用不同功



能模块的脚本文件。关键代码如下：

```
<?php
	switch($lmbs){						//获取变量$lmbs 的值
		case "车源信息查询":				//判断当变量$lmbs 的值是"车源信息查询"时输出下面的内容
			include("car_select.php");		//调用指定的包含文件
		break;							//跳出循环
	//这里省略了部分代码
		case "":						//判断当变量的值为空时
			include("car_select.php");		//调用该文件
		break;							//跳出循环
	}
?>
```

5.6　车源信息查询模块设计

5.6.1　车源信息查询模块概述

车源信息查询模块的功能是根据客户提供的货物配送的地点，从数据库中查询有关该路线的车辆，为客户提供一个合理配送的路线，最后确定发货订单。操作流程如图 5.23 所示。

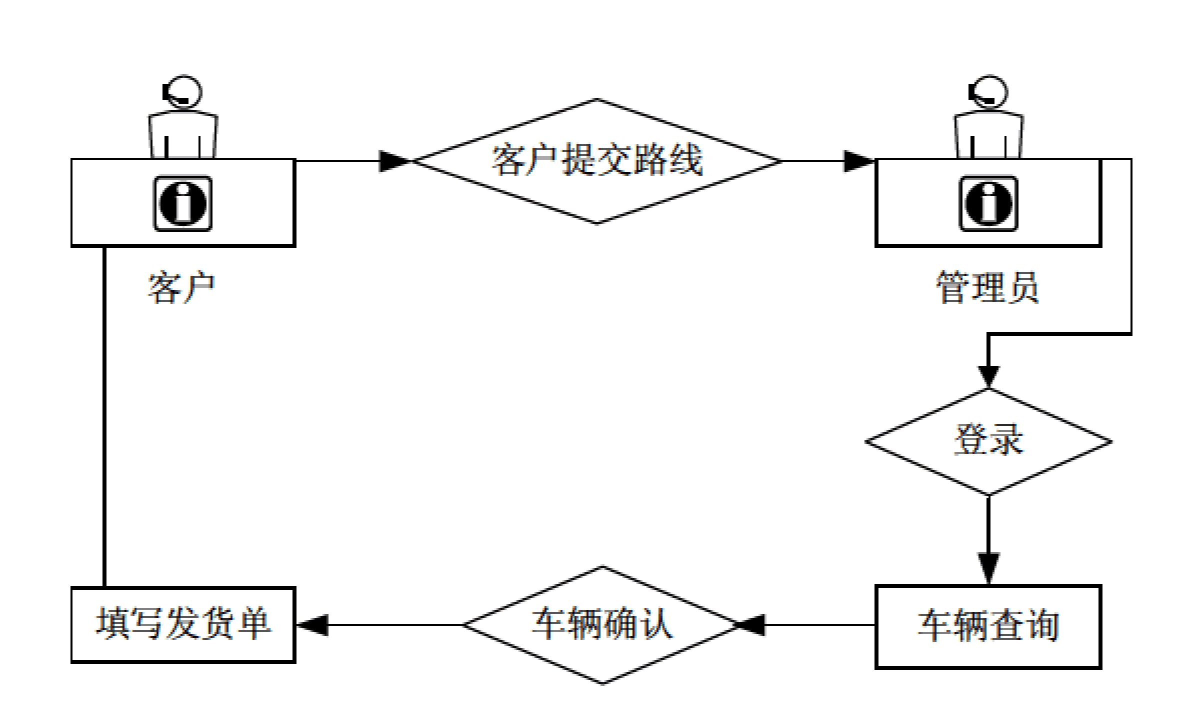


图 5.23　车源信息查询的操作流程

5.6.2　车源信息查询模块技术分析

车源信息查询模块的主要功能就是查询车辆的使用信息，为客户提供最合适的货物配送路线。其中应用的关键技术自然就是查询方法，为了给客户提供最合适、最满意、最快捷的配送服务，这里使用的是模糊查询技术。

通过模糊查询技术，只要客户提出配送货物的出发地点和到达地点，管理员就可以从数据库中提取出所有与该路线相关的车辆信息，包括车辆的承载能力、车辆是否被占用和车辆的使用情况，客户可以根据实际的情况（时间、货物数量、路线等）选择车辆，确认后填写发货单。

模糊查询技术的实现是在 car_select.php 文件中完成的，首先要与数据库建立连接；然后创建一个 form 表单，设置两个文本框 select1 和 select2 用于提交出发地点和到达地点，将表单的值提交到当前页；最后编写 PHP 脚本语句，以表单中获取的数据为条件，执行 like 模糊查询，从数据库的指定表中查询符合条件的信息，并且将查询出的信息显示到页面中。程序的关键代码如下：

例程02　代码位置：光盘\TM\05\wlgl\car_select.php

```
<?php
 session_start(); include("conn/conn.php");			//连接数据库
 ?>
<!-- ----------------------------------创建 form 表单，提交出发地点和到达地点---------------------------------------- -->
```

```
<form id="form1" name="form1" method="post" action="indexs.php?lmbs=车源信息查询" onSubmit="javascript:
return check_select_car();">
    <td height="30" colspan="4" align="right" bgcolor="#FFFFFF">查询
❶          <input name="select1" type="text" id="select1" size="10">至
❷          <input name="select2" type="text" id="select2" size="10"></td>
    <td colspan="2" bgcolor="#FFFFFF"><input type="submit" name="Submit" value="提交" /></td>
    </form>
<!-- ------------------获取表单提交的值，根据该条件执行 like 模糊查询，从数据库中读取数据------------------ -->
<?php
❸   if($select1==true || $select2==true){           //判断表单提交的值是否为真
     /* *******************以表单提交的值为条件，从数据表中查询数据***************** */
❹   $query="select * from tb_car where car_road like '%$select1%' and car_road like '%$select2%'";
     $result=mysql_query($query);                    //执行查询语句
     if(mysql_num_rows($result)>0){                  //判断是否存在符合条件的数据
     while($myrow=mysql_fetch_array($result)){       //应用 while 循环语句，输出符合条件的数据
?>
<tr>
     <td height="26" align="center" bgcolor="#FFFFFF"><?php echo $myrow[car_number];?></td>
    <td bgcolor="#FFFFFF"><?php echo $myrow[car_road];?></td>
    <td colspan="2" bgcolor="#FFFFFF"><?php echo $myrow[car_content];?></td>
    <td bgcolor="#FFFFFF">
     <?php
          /* **********根据车牌号码，从车辆日志信息表中读取对应车辆的使用信息********** */
          $querys=mysql_query("select * from tb_car_log where car_number='$myrow[car_number]'");
          $myrows=mysql_fetch_array($querys);
          echo $myrows[car_log];                     //输出车辆的使用日志信息
     ?>
     </td>
    <td align="center" bgcolor="#FFFFFF">
     <?php
     if($myrows[car_log]==null){                     //判断车辆日志是否为空，如果为空，则执行下面的内容
          echo "<a href='indexs.php?lmbs=发货单&ljid=$myrow[id]'>使用该车</a>";
     }else{                                          //如果车辆日志不为空，则执行下面的内容
          echo "<a href='indexs.php?lmbs=发货单&ljid=$myrow[id]'>预定该车</a>";
     }
     ?>
     </td>
</tr>
<?php }}else{   echo "您查找的路线不存在!"; }}?>
```

代码贴士

❶ select1：指定出发地点。

❷ select2：指定到达地点。

❸ if 语句：判断从表单中获取的变量是否为真。

❹ like：执行模糊查询，从数据库中读取符合条件的数据。

5.6.3 车源信息查询模块的实现过程

车源信息查询模块使用的数据表：tb_car、tb_car_log

车源信息查询模块的功能是根据客户提供的信息，从数据库中读取有关某个路线的车辆使用的情



形，及时为客户确定一个合理的物流配送方案。该模块的运行结果如图 5.24 所示。



图 5.24　车源信息查询模块的运行结果

车源信息查询功能的实现通过 car_select.php 文件来完成，首先与数据库建立连接；然后创建一个 form 表单，将客户的地址信息提交到当前页，作为查询的条件；最后编写 PHP 脚本语句，以表单中获取的地点数据为条件，执行 like 模糊查询，从数据库的指定表中查询符合条件的信息，并且将查询出的信息显示到页面中。其关键代码可以参考 5.6.2 节的例程 02。

5.6.4　单元测试

在对车源信息查询模块进行测试的过程中，当输入指定的出发点和到达地点进行提交后，出现如图 5.25 所示的页面，程序出错，没有找到符合条件的记录。

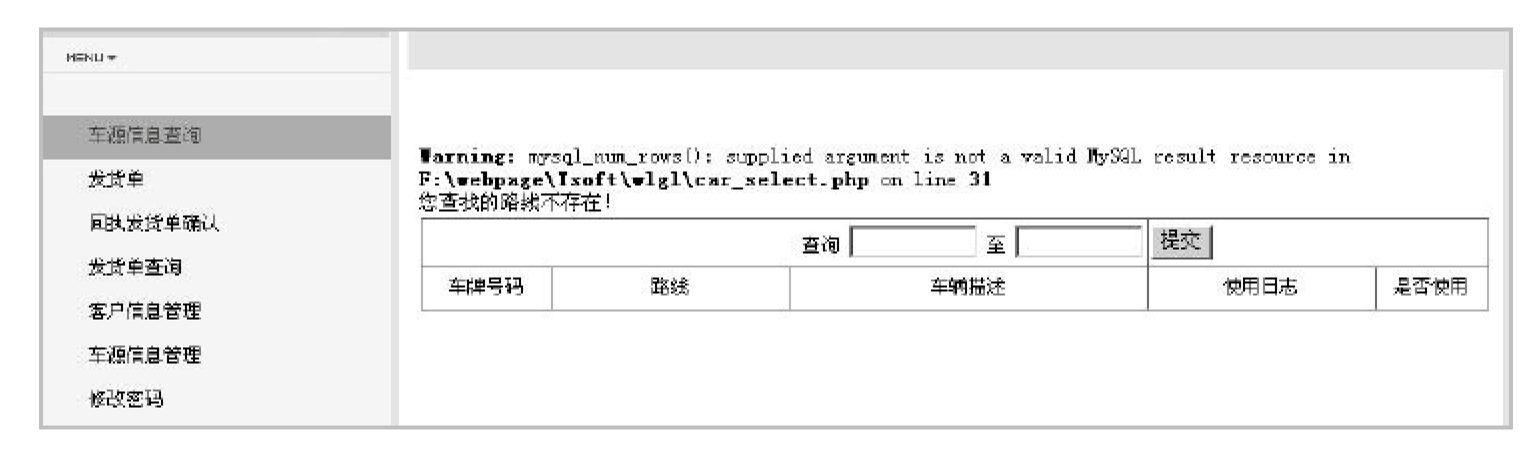


图 5.25　车源信息查询的错误运行结果

分析错误原因：从弹出的错误信息来看，错误应该与数据库有关。下面从数据库开始查找错误的原因，首先确定程序中使用的数据库、数据表和字段的书写是否正确，仔细核对没有发现问题；然后确认使用的 select 查询语句的格式是否正确，经检查发现，在编写 select 查询语句时使用了错误的语句格式，在执行 like 模糊查询时没有将参数的两侧加上单引号（car_road like %$select1% and car_road like %$select2%），从而导致程序中类型不匹配的错误。其错误语句和正确语句的对比如图 5.26 所示。

```
<?php
    if($select1==true || $select2==true){
        $query="select * from tb_car where car_road like %$select1% and car_road like %$select2%";
        $result=mysql_query($query);
    if(mysql_num_rows($result)>0){
        while($myrow=mysql_fetch_array($result)){
?>

<?php
    if($select1==true || $select2==true){
        $query="select * from tb_car where car_road like '%$select1%' and car_road like '%$select2%'";
        $result=mysql_query($query);
    if(mysql_num_rows($result)>0){
        while($myrow=mysql_fetch_array($result)){
?>
```


图 5.26　错误语句和正确语句对比

正确的格式为 car_road like '%$select1%' and car_road like '%$select2%'，或者 car_road like "%".$select1."%' and car_road like '%".$select2."%'。修改后保存程序，程序正常运行。

注意 在执行 SQL 语句的过程中，传入 SQL 语句的参数为字符型数据，所以在传入参数的两侧应该加上单引号，表示该数据为字符型，否则程序将发生类型不匹配的错误。

说明 在 PHP 中的字符串有两种表示格式：第一种是将字符串用单引号括起来；第二种是将字符串用双引号括起来。如果采用第二种格式，PHP 变量可以直接写在该字符串中，系统不会将该变量看作字符串的一部分，而是将其看作 PHP 的合法变量。

5.7　发货单管理模块设计

5.7.1　发货单管理模块概述

发货单管理模块主要包括发货单的填写、发货单查询、发货单打印和发货单删除。发货单管理模块的主要功能如图 5.27 所示。

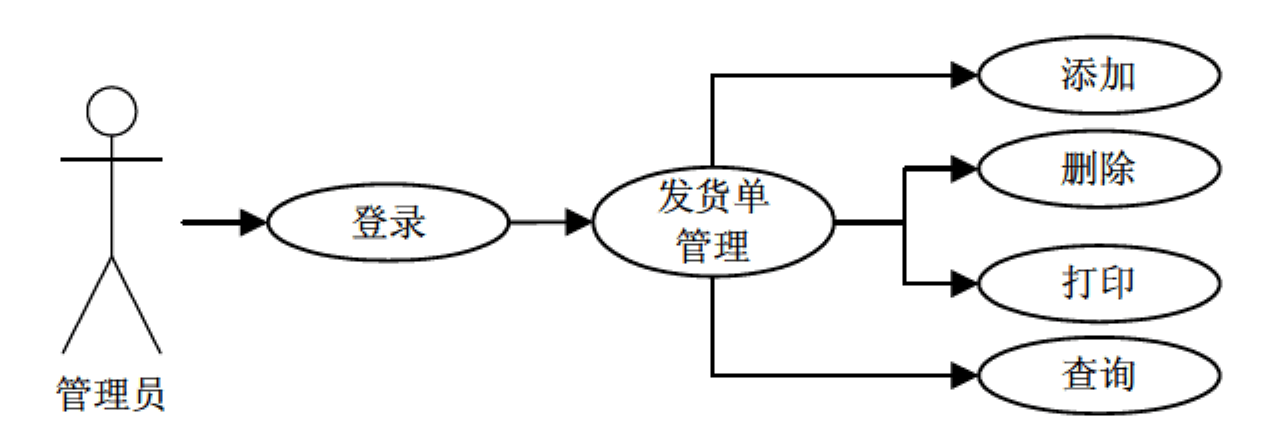


图 5.27　发货单管理模块的功能结构图

5.7.2　发货单管理模块技术分析

如何才能发挥函数的最大作用？设计者不但要能够从表面理解函数的功能和作用，而且要能够以一种发散的思维方式去考虑问题，不要被其本身的功能所束缚，要将函数中某些功能有机地结合起来，从而发挥其更大的作用。

在发货单管理模块中应用到一个发货单的编号，该编号是发货单的唯一标识，不允许存在重复。在编写生成这个编号的程序时，首选方案是应用随机函数获取随机的数字作为编号，这是一个最直接的做法，但是考虑到发货单编号不但具有唯一性，而且还要使其具有一定的标志性，从这个角度考虑使用随机函数就有些不适合，因为其不具备一定的标志性。这时，如果应用 date()函数，则可以达到上述所说的要求，不但编号具有唯一性，而且还有规律可循。

为了能够更好地理解这种发散思维的编程思想，首先介绍一下 date()函数。语法格式如下：

```
string date ( string format , int timestamp)
```

该函数返回将参数 timestamp 按照指定格式格式化而产生的字符串，其中参数 timestamp 是可选的，默认值为 time()，即如果没有给出时间戳，则使用本地当前时间。

date()函数的参数 format 的格式化选项如表 5.1 所示。

表 5.1 参数 format 的格式化选项

参　数	说　明
a	小写的上午和下午值，返回值为 am 或 pm
A	大写的上午和下午值，返回值为 AM 或 PM
B	Swatch Internet 标准时，返回值为 000～999
d	月份中的第几天，有前导零的 2 位数字，返回值为 01～31
D	星期中的第几天，文本格式，3 个字母，返回值为 Mon～Sun
F	月份，完整的文本格式，返回值为 January～December
g	小时，12 小时格式，没有前导零，返回值为 1～12
G	小时，24 小时格式，没有前导零，返回值为 0～23
h	小时，12 小时格式，有前导零，返回值为 01～12
H	小时，24 小时格式，有前导零，返回值为 00～23
i	有前导零的分钟数，返回值为 00～59
I	判断是否为夏令时，如果是，返回值为 1，否则为 0
j	月份中的第几天，没有前导零，返回值为 1～31
l	星期数，完整的文本格式，返回值为 Sunday 到 Saturday
L	判断是否为闰年，如果是，返回值为 1，否则为 0
m	数字表示的月份，有前导零，返回值为 01～12
M	3 个字母缩写表示的月份，返回值为 Jan～Dec
n	数字表示的月份，没有前导零，返回值为 1～12
o	与格林威治时间相差的小时数，例如 0200
r	RFC 822 格式的日期，例如 Thu，21 Dec 2000 16：01：07 +0200
s	秒数，有前导零，返回值为 00～59
S	每月天数后面的英文后缀，2 个字符，例如 st、nd、rd 或者 th。可以和 j 一起使用
t	指定月份所应有的天数
T	本机所在的时区
U	从 UNIX 纪元（January 1 1970 00:00:00 GMT）开始至今的秒数
w	星期中的第几天，数字表示，返回值为 0～6
W	ISO-8601 格式年份中的第几周，每周从星期一开始

续表

参　数	说　明
Y	4 位数字完整表示的年份，返回值如 1998、2008
y	2 位数字表示的年份，返回值如 88 或 08
z	年份中的第几天，返回值为 0～366
Z	时差偏移量的秒数。UTC 西边的时区偏移量总是负的，UTC 东边的时区偏移量总是正的，返回值为 -43200～43200

注意 这里有效的时间戳典型范围是格林威治时间 1901 年 12 月 13 日 20:45:54 到 2038 年 1 月 19 日 03:14:07（此范围符合 32 位有符号整数的最小值和最大值）。在 Windows 系统中此范围限制为从 1970 年 1 月 1 日到 2038 年 1 月 19 日。

技巧 要应用发散的思维方式，在考虑编程方法时就不能被 date()函数自身的功能所限制，date()函数获取的时间不只可以用来记录历史中的某一时刻，而且可以将其应用到其他的地方，作为一个有规律的随机数。

在本项目中的发货单填单模块中就是应用 date()函数获取当前时间和车辆 ID 作为发货单的编号。关键代码如下：

```
<?php
echo date("YmdHis");                                    //获取系统的当前时间
echo $ljid;                                             //获取车辆的 ID
?>"
```

其运行结果如图 5.28 所示。

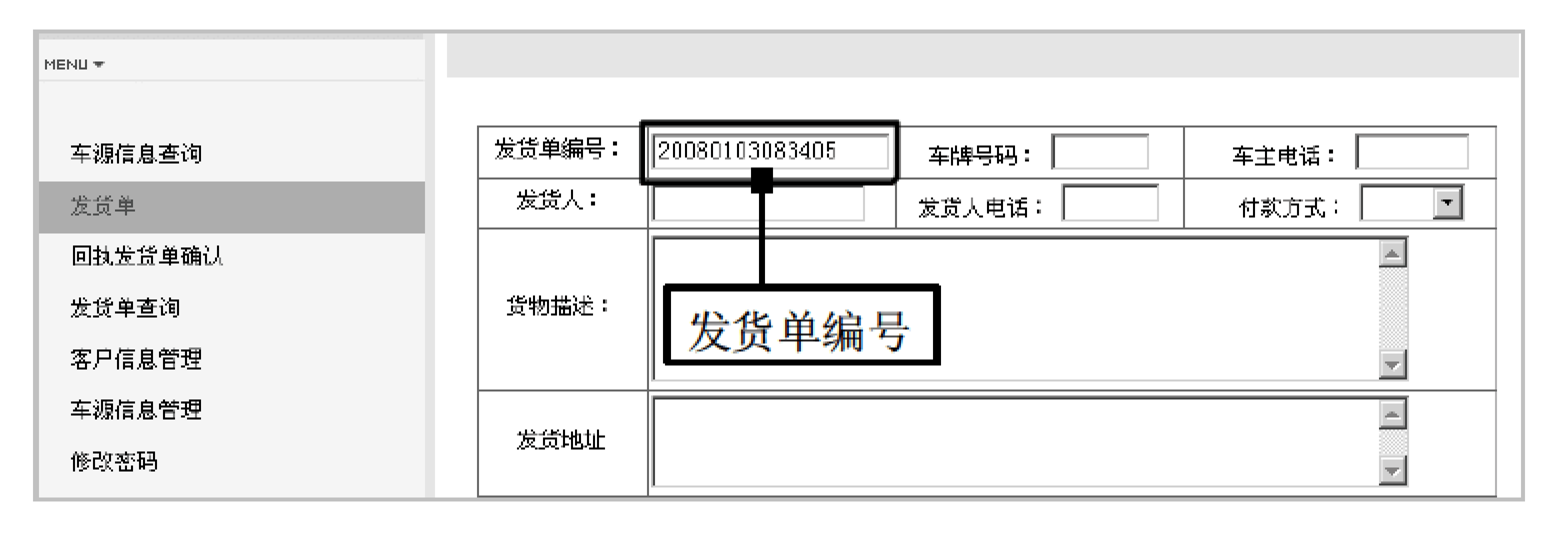


图 5.28　发货单编号生成的运行结果

技巧 程序开发不只是要看到其表面的功能和属性，更重要的是将某些功能或者属性有机地结合起来，从而开发出更有意义的功能，这才是程序员所要追求的。例如，在学习字符串中的 str_ireplace()函数时，不但要知道该函数的功能是实现字符串的替换功能，而且要学会如何巧妙地运用其实现更好的功能。当换个角度去考虑这个函数的功能时，对这个函数的参数字符串进行颜色改变。这时实现的功能就不只是简单的字符串替换，而且对字符串实现了描红的功能。



5.7.3　发货单填单的实现过程

发货单填单模块使用的数据表： tb_customer、tb_car_log、tb_shopping

发货单的填写是客户在确定物流配送路线以后填写的一个货物配送详细信息单据，内容包括发货单编号、车牌号码、车主电话、发货人、发货人电话、发货地址、付款方式、货物描述、收货人、收货人电话、收货地址和说明。发货单填单的运行结果如图 5.29 所示。



图 5.29　发货单填单的运行结果

发货单填单的实现通过 insert_dds.php 文件来完成，首先随机生成一个订单编号，然后创建一个 form 表单，最后将表单中的数据提交到数据库中。在 insert_dds.php 文件中使用的重要表单元素如表 5.2 所示。

表 5.2　发货单填单中使用的重要表单元素

名　称	元素类型	重要属性	含　义
form1	form	method="post" action="insert_dd_ok.php" onSubmit="javascript: return check_dd()"	表单
fahuo_id	text	id="fahuo_id" value="<?php echo date("YmdHis");?><?php echo $ljid;?>"	发货单编号
car_number	text	value="<?php echo $myrow[car_number]?>"	车牌号码
car_tel	text	value="<?php echo $myrow[tel];?>"	车主电话
fahuo_fk	select	<option selected="selected"></option><option>发货人付款</option><option>第三方付款</option>	付款方式
car_log	textarea	cols="65" rows="5"	说明
Submit	submit	value="提交"	提交表单

将表单中提交的数据存储到数据库是通过 insert_dd_ok.php 文件来完成的，在该文件中首先连接数据库，然后通过 preg_match()函数判断表单中提交的电话号码格式是否正确，最后将表单中提交的数据存储到 tb_shopping 表中，并且将车辆使用的说明提交到 tb_car_log 表中，将客户信息提交到 tb_customer 表中。关键代码如下：

例程03　代码位置：光盘\TM\05\wlgl\insert_dd_ok.php

```
<?php
if($_SESSION[admin_user]==true){                                //判断登录的用户是否正确
```



```
include("conn/conn.php");                                    //连接数据库
$fahuo_time=date("Y-m-d H:i:s");                             //获取系统当前时间
if($Submit==true){                                           //如果提交表单，则执行下面的语句
    /* **********************判断表单中提交的电话号码格式是否正确********************** */
    if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$car_tel,$countes)){
        if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$fahuo_tel,$countes)){

if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$shouhuo_tel,$countes)){
    /* **********************将表单提交的数据添加到 tb_shopping 表中************************ */
    $query="insert into tb_shopping (car_number,car_tel,fahuo_id,fahuo_user,
fahuo_tel,fahuo_address,fahuo_content,fahuo_time,fahuo_fk,shouhuo_user,shouhuo_address,shouhuo_tel)val
ues('$car_number','$car_tel','$fahuo_id','$fahuo_user','$fahuo_tel','$fahuo_address','$fahuo_content','$fahuo_ti
me','$fahuo_fk','$shouhuo_user','$shouhuo_address','$shouhuo_tel')";
    $result=mysql_query($query);
    /* ************************将车辆使用说明添加到 tb_car_log 表中* *********************** */
    $querys="insert into
tb_car_log(car_log,car_number,log_date,fahuo_id)values('$car_log','$car_number','$fahuo_time','$fahuo_id')";
    $results=mysql_query($querys);
    /* ************************将客户信息提交到 tb_customer 表中* ************************** */
    $queryss="insert into tb_customer
(customer_user,customer_tel,customer_address)values('$fahuo_user','$fahuo_tel','$fahuo_address')";
    $resultss=mysql_query($queryss);
    echo "<script>alert('发货单添加成功!');window.location.href='indexs.php?lmbs=发货单';</script>";
            }else{
              echo "<script>alert('您输入的收货人的电话号码格式不正确!!');history.back()</script>"; }
        }else{
            echo "<script>alert('您输入的发货人的电话号码格式不正确!!');history.back()</script>"; }
    }else{
          echo "<script>alert('您输入的车主的电话号码格式不正确!!');history.back()</script>"; }
}?>
<?php }else{                                                 //判断用户是否正确登录，如果不是则返回登录页面
    echo "<script>alert('请您正确登录!'); window.location.href='index.php';</script>";}
?>
```

5.7.4 发货单查询的实现过程

*发货单查询模块使用的数据表：*tb_car_log、tb_shopping

发货单查询是为了便于对发货单进行查找以及处理而设计的一个功能，通过其可以准确地查找到指定的发货单，并且还设置一个发货单删除的功能，可以对已经失效或者作废的发货单进行删除。其实现的原理主要是通过以发货单编号为条件的精确查找或者以发货人姓名为条件的模糊查询，然后将查询的结果输出到页面中。运行结果如图 5.30 所示。



图 5.30　发货单查询的运行结果

发货单查询的实现主要通过 hwys.php 文件



完成，首先与数据库建立连接，然后创建一个表单 form1，通过该表单来提交查询的条件，可以选择精确查找（以发货单编号为条件）或者模糊查询（以发货人姓名为条件），最后根据 form1 表单中提交的数据，执行查询语句，从数据库中读取出符合条件的发货单的内容。程序的关键代码如下：

例程04　代码位置：光盘\TM\05\wlgl\hwys.php

```
<?php session_start(); include("conn/conn.php");                    //初始化 session 变量，连接数据库
      if($_SESSION[admin_user]==true){                                //判断管理员是否是正确登录
?>
<!--省略了部分代码-->
<?php
      /* **************判断当 form1 表单提交的值为发货单编号时，执行下面的语句* ************* */
      if($select1=="发货单编号" and $select!=""){
      $query="select * from tb_shopping where fahuo_id='$select'";
      $result=mysql_query($query);                                    //执行查询语句
      $myrow=mysql_fetch_array($result);                              //获取查询结果
      }
      /* *****************当 form1 表单提交的值为发货人时，执行下面的语句* **************** */
      if($select1=="发货人" and $select!=""){
      $query="select * from tb_shopping where fahuo_user='$select'";
      $result=mysql_query($query);                                    //检索指定发货人的信息
      $myrow=mysql_fetch_array($result);
      }
      ?>
…                                                                     //省略部分 HTML 代码
<td colspan="4" bgcolor="#FFFFFF"><textarea name="car_log">
<!-- ------------------------------------输出车辆日志中的车辆使用情况----------------------------------------- -->
<?php
      $querys="select * from tb_car_log where fahuo_id='$myrow[fahuo_id]'"; //从车辆日志表中查找指定的数据
      $results=mysql_query($querys);                                  //执行查询语句
      $myrows=mysql_fetch_array($results);                            //获取查询结果
      echo $myrows[car_log];                                          //输出车辆日志信息
      ?>
      </textarea></td>
      <!-- --------------------------根据发货单的 ID 设置一个发货单删除的超链接-------------------------- -->
      <td bgcolor="#FFFFFF"><a href="fhd_qr.php?delete=<?php echo $myrow[id];?>">删除发货单</a></td>
…                                                                     //省略部分 HTML 代码
<?php }else{    //判断管理员是否正确登录，如果错误则返回到登录页
      echo "<script>alert('请您正确登录!'); window.location.href='index.php';</script>";
}
?>
```

发货单删除的操作通过 fhd_qr.php 文件来完成，主要根据超链接中提供的发货单 ID 执行删除的操作。关键代码如下：

例程05　代码位置：光盘\TM\05\wlgl\fhd_qr.php

```
<?php session_start(); include("conn/conn.php");                    //初始化 session 变量，连接数据库
if($_SESSION[admin_user]==true){
      if($delete==true){                                              //判断提交的发货单 ID 是否为真
            $query="delete from tb_shopping where id='$delete'";      //根据发货单 ID 执行删除发货单的操作
            $result=mysql_query($query);
```

```
            /* *********************发货单删除成功，弹出提示，并返回到上一页********************* */
            echo "<script>alert('发货单删除成功!');window.location.href='indexs.php?lmbs=发货单查询';</script>";
        }
}else{
        /* **********************发货单删除失败，弹出提示，并返回到上一页*********************** */
        echo "<script>alert('请您正确登录!'); window.location.href='index.php';</script>";
}
?>
```

5.7.5 发货单打印的实现过程

发货单的打印是为货物的发送提供一个书面的依据，作为企业与客户商业活动的凭证，其内容必须真实、准确。发货单打印的运行结果如图 5.31 所示。



图 5.31 发货单打印的运行结果

这里的发货单打印技术主要通过框架打印技术来完成，即只打印框架中的内容，而网页中其他的内容不打印。实现原理是：首先在 insert_dd.php 文件中创建一个浮动框架，设置要指定打印内容的范围，并且连接到要打印的 insert_dds.php 文件，然后在指定的文件中输出要打印的内容，最后应用 onClick 事件调用 parent.content.focus()和 window.print()方法实现打印功能。

在 insert_dd.php 文件中创建一个浮动框架，设置框架名称为 content，设置打印的范围宽 97%、高 252%，连接到要打印的 insert_dds.php 文件，程序代码如下：

例程06 代码位置：光盘\TM\05\wlgl\insert_dd.php

```
        <iframe name="content" src="insert_dds.php?ljid=<?php echo $ljid;?>"   frameborder="0" width="97%"
height="252%"></iframe>
```

设置要打印的内容，通过 insert_dds.php 文件来完成，发货单填写的内容以表单的形式显示，其中的内容可以参考光盘中的 TM\05\wlgl\insert_dds.php 文件，这里不再赘述。

最后回到 insert_dd.php 页中，在该页中设置一个超链接，应用 onClick 事件调用 parent.content.focus()和 window.print()方法实现发货单的打印功能。程序关键代码如下：

例程07 代码位置：光盘\TM\05\wlgl\insert_dd.php

```
<a href="#"
        onClick="parent.content.focus();window.print();">
        <img src="images/dl_033.gif" width="87" height="27" border="0">
</a>
```

5.7.6 单元测试

在进行发货单填单模块的测试过程中，当打开发货单填单模块的页面时，其运行结果如图 5.32 所示。



在该页面中整个浮动框架中的内容没有显示。分析错误原因，应该是在设置浮动框架时出现的问题，查看 insert_dd.php 文件中的代码，发现原来是设置浮动框架时指定的链接文件出现问题，使用了错误的链接文件名称。链接文件的正确名称是 insert_dds.php，而程序中将该文件的名称书写为 inser_ddes.php，从而导致页面中没有输出任何的内容。将 insert_dd.php 文件中的内容进行修改，重新运行程序，运行结果如图 5.33 所示。

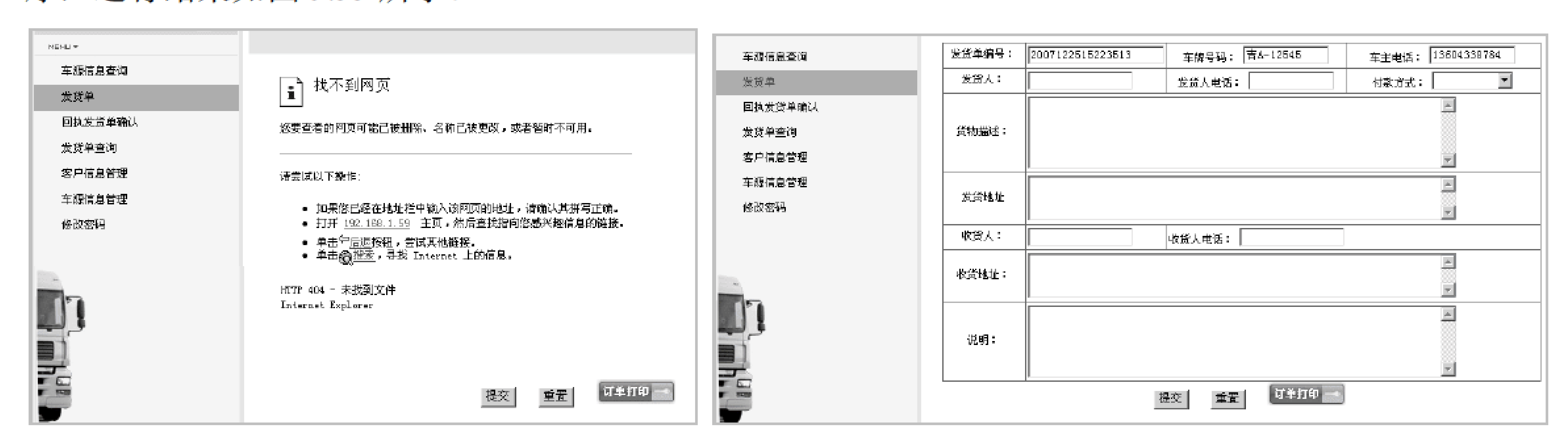


图 5.32　发货单填单模块的运行结果　　图 5.33　发货单模块的运行结果

5.8　回执单验收管理模块设计

5.8.1　回执单模块概述

回执单模块的主要功能就是对货物配送完成确认：将该发货单执行类型更新为“1”，表明该次物流配送已经完成；清空该车辆的使用日志，便于执行下一个订单。该模块的业务操作流程如图 5.34 所示。

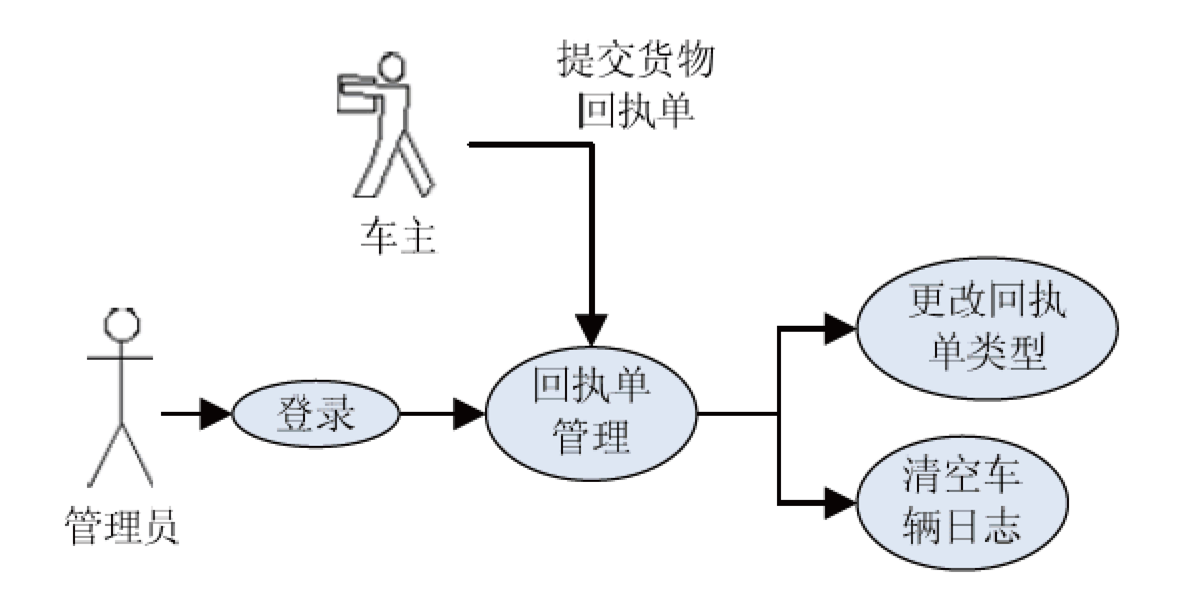


图 5.34　回执单模块的业务操作流程

5.8.2　回执单模块技术分析

在回执单模块中，要对回执单进行确认首先要查找到指定的发货单，并且对其内容进行核实。这里应用的是根据发货单的编号进行查询，从数据库中读取出符合条件的数据。在执行查询语句、从数据库中读取数据时，主要应用 mysql_query()、mysql_num_rows()和 mysql_fetch_array()函数。

首先通过 mysql_query()函数执行一个 select 查询语句，然后通过 mysql_num_rows()函数判断查询

的结果是否有值，如果没有值则输出“您查找的发货单编号不存在！”，如果有值则执行 mysql_fetch_arrau()函数，输出数据库中查询到的数据。

mysql_query()函数向与指定的连接标识符关联的服务器中的当前活动数据库发送一条 MySQL 查询。语法如下：

```
resource mysql_query ( string query [, resource link_identifier] )
```

mysql_query()函数的参数 query 是字符串的类型，传入的是 SQL 指令；参数 link_identfier 是资源类型，传入的是由 mysql_connect()函数或者 mysql1_pconnect()函数返回的连接号。如果省略该参数，则会使用最后一个打开的 MySQL 数据库连接。

注意 查询字符串不应以分号结束。

mysql_num_rows()函数获取结果集中行的数目。语法如下：

```
int mysql_num_rows ( resource result )
```

注意 该函数仅对 SELECT 语句有效。

mysql_fetch_array()函数返回根据从结果集获取的行生成的数组，如果没有更多行则返回 false。语法如下：

```
array mysql_fetch_array ( resource result [, int result_type] )
```

mysql_fetch_array()函数的参数说明如表 5.3 所示。

表 5.3 mysql_fetch_array()函数的参数说明

参 数 名 称	说　明
result	资源类型的参数，要传入的是由 mysql_query()函数返回的“数据指针”
result_type	可选项，整数型参数，要传入的是 MYSQL_ASSOC、MYSQL_NUM 和 MYSQL_BOTH 3 种由 PHP 定义好的常数之一，默认值是 MYSQL_BOTH 用 MYSQL_ASSOC 只得到关联索引（相当于 mysql_fetch_assoc()函数） 用 MYSQL_NUM 只得到数字索引（相当于 mysql_fetch_row()函数） 用 MYSQL_BOTH 将得到一个同时包含关联和数字索引的数组

注意 该函数返回的字段名是区分大小写的。

通过上述 3 个函数就可以完成从数据库中读取数据，并且将数据输出到页面的功能。

注意 使用上述方法输出的只是数据库中的一条数据。如果加上 while 循环语句，就可以完成数据库中所有符合条件数据的循环输出。





应用 while 循环语句读取数据库中的数据，这里以读取 tb_shopping 表中的数据为例，程序代码如下：

```
<?php include("conn/conn.php");                              //连接数据库
    $query="select * from tb_shopping ";                      //编写查询语句
    $result=mysql_query($query);                              //执行查询语句
❶   if(mysql_num_rows($result)<1){                            //判断数据库中是否存在数据
    echo "您查找的内容不存在!";
    }else{
❷   while($myrow=mysql_fetch_array($result)){                 //循环输出数据库中的数据
?>
<?php
 echo $myrow[id];                                             //输出数据库中的数据
?>
…                                                             //输出内容的代码部分略
<?php }?>
```

代码贴士

❶ mysql_num_rows()：返回根据从结果集获取的行生成的数组，如果没有更多行则返回 false。

❷ mysql_fetch_array()：获取结果集中行的数目。该函数只对 SELECT 语句有效。

5.8.3　回执单验收模块的实现过程

回执单验收模块使用的数据表：tb_car_log、tb_shopping

回执单验收模块的主要功能就是对货物配送完成确认，运行结果如图 5.35 所示。

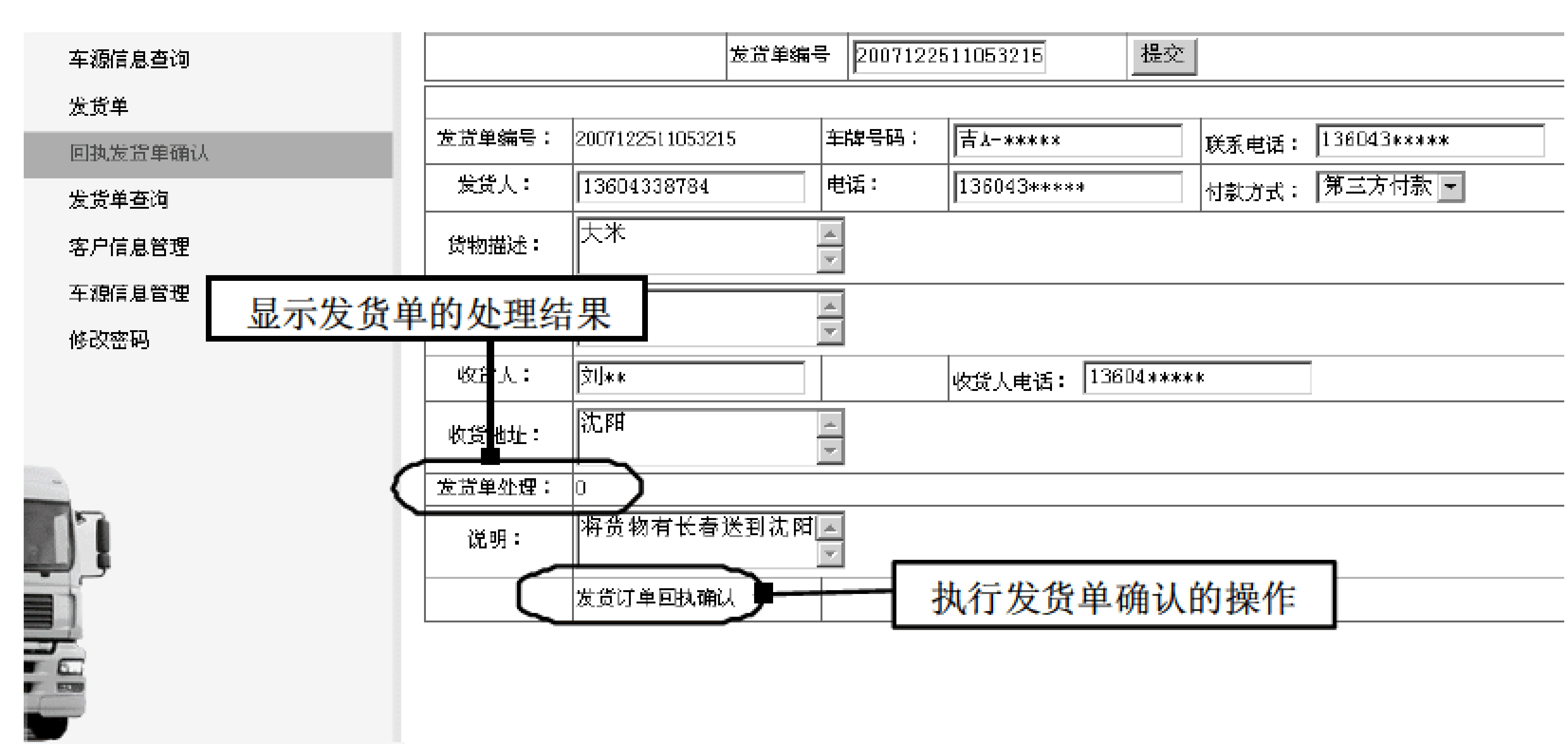


图 5.35　回执单验收模块的运行结果

回执单验收模块的实现过程是：首先在 select_dhd.php 页中根据发货单的编号查询出指定的发货单，对发货单进行核实，然后单击“发货订单回执确认”超链接，实现回执单的验收。关键代码如下：

例程08　代码位置：光盘\TM\05\wlgl\select_dhd.php

```
<?php
    include("conn/conn.php");                                      //连接数据库
    if($Submit==true){
        $query="select * from tb_shopping where fahuo_id='$select'";   //编写查询语句
        $result=mysql_query($query);                               //执行查询语句
        if(mysql_num_rows($result)<1){                             //判断数据库中是否存在数据
```

```
            echo "您查找的发货单编号不存在!";
        }else{
            $myrow=mysql_fetch_array($result);                          //输出数据库中的数据
?>
…                                                                        //输出内容的代码部分略
    <td colspan="4" bgcolor="#FFFFFF">
        <textarea name="car_log">
            <?php
                /* ***************根据提交的货物编号，查询车辆日志中的对应日志************** */
                $querys="select * from tb_car_log where fahuo_id='$myrow[fahuo_id]'";
                $results=mysql_query($querys);                          //执行查询语句
                $myrows=mysql_fetch_array($results);                    //输出数据库中的数据
                echo $myrows[car_log];                                  //输出数据库中的数据
            ?>
        </textarea></td>
    …                                                                    //输出内容的代码部分略
    <!-- ------------------------------------设置超链接，执行发货单的确认---------------------------------- -->
    <td bgcolor="#FFFFFF"><a href="fhd_qr.php?fahuo_id=<?php echo $myrow[fahuo_id];?>">发货订单回执
确认</a></td>
```

最后在 fhd_qr.php 页根据超链接提供的发货单编号更新发货单类型为 1，清空对应车辆的使用日志。程序的关键代码如下：

例程09　代码位置：光盘\TM\05\wlgl\fhd_qr.php

```
<?php session_start(); include("conn/conn.php");                    //初始化 session 变量
if($_SESSION[admin_user]==true){                                     //判断用户是否是正确登录
    if($_GET[fahuo_id]==true){                                       //判断超链接传递的变量值是否为真
        /* *****************以超链接传递的变量值为条件，编写删除语句******************* */
        $query="delete from tb_car_log where fahuo_id='$fahuo_id'";
        $result=mysql_query($query);                                 //执行删除语句
        /* ********************更新发货单数据表中发货的类型为 1*********************** */
        $query="update tb_shopping set fahuo_ys='1' where fahuo_id='$fahuo_id' ";
        $result=mysql_query($query);
            echo  "<script>alert('发货回执单处理成功!');window.location.href='indexs.php?lmbs=回执发货单
确认';</script>"; }
    }else{
        echo "<script>alert('请您正确登录!'); window.location.href='index.php';</script>";
    }
?>
```

5.9　基础信息管理模块设计

5.9.1　基础信息管理模块概述

基础信息管理模块主要包括客户信息管理，用于添加客户信息和删除客户信息；车源信息管理，主要用于对车源信息进行管理，实现车源信息的添加、修改和删除功能；管理员信息管理，用于修改管理员的密码。基础信息管理模块的功能结果如图 5.36 所示。





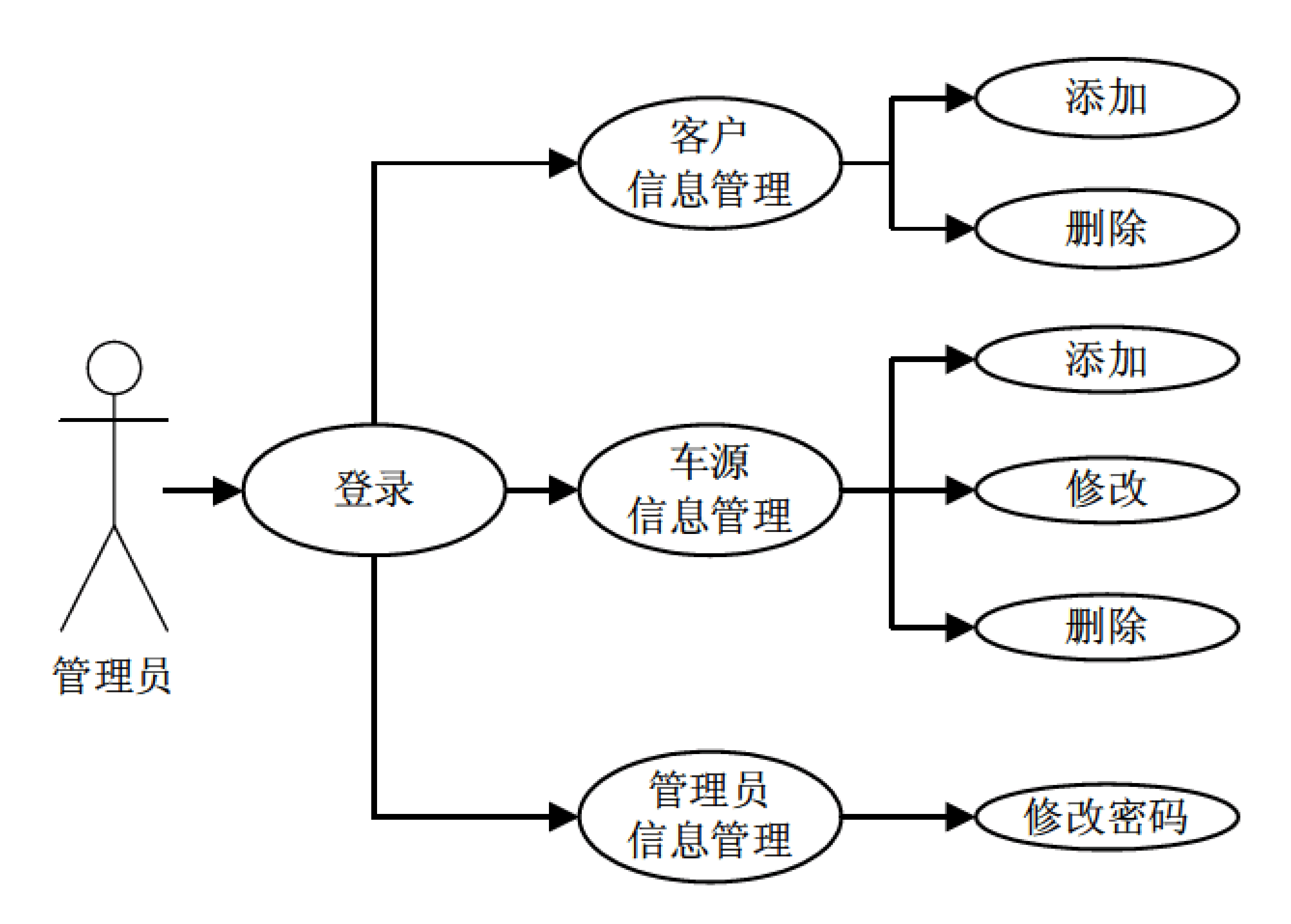


图 5.36　基础信息管理模块的功能结构图

5.9.2　管理员信息管理模块技术分析

修改管理员密码的模块中，在对密码进行处理的过程中用的是面向对象技术实现密码的修改。

面向对象技术的关键就是类的创建，首先创建一个 chkuser 类，其中定义 5 个数据成员，实现对管理员密码的修改，分别是管理员名称（$admin_user）、管理员密码（$admin_pass）、管理员新密码（$admin_new_pass）、管理员新密码确认（$admin_new_pass2）和验证码（$yzm），然后应用 chkuser()函数实现数据成员的初始化，最后应用数据成员函数 chkinput()实现密码的修改。程序代码如下：

例程10　代码位置：光盘\TM\05\wlgl\update_pass_ok.php

```
<?php	session_start();	//初始化 session 变量
❶	class chkuser{	//创建一个 chkuser 类
		var $admin_user;	//创建数据成员，用户名
		var $admin_pass;	//创建数据成员，密码
		var $admin_new_pass;	//创建数据成员，新密码
		var $admin_new_pass2;	//创建数据成员，密码确认
		var $yzm;	//创建数据成员，验证码
	function chkuser($x,$y,$m,$n,$z){	//创建 chkuser()函数，实现数据成员初始化
		$this->admin_user=$x;
		$this->admin_pass=$y;
		$this->admin_new_pass=$m;
		$this->admin_new_pass2=$n;
		$this->yzm=$z;
	}
❷	function chkinput(){	//创建 chkinput()函数，实现密码更改
		//判断输入的验证码是否正确
❸		if(strval($this->yzm)!=$_SESSION["autonum1"]){
			echo "<script>alert('验证码输入错误!');history.back();</script>";
			exit;
		}
	include("conn/conn.php");	//连接数据库
		//判断输入的密码是否正确
		$query=mysql_query("select * from tb_admin where admin_user='".$this->admin_user."' and
```

```
admin_pass='".$this->admin_pass."'");
                if(mysql_num_rows($query)<1){
                    /* *********************如果密码不正确，则提示重新输入密码********************* */
                    echo "<script>alert('您输入的密码不正确，请重新输入');history.back();</script>";
                }else{
                    /* *****************如果密码正确，则应用更新语句，实现密码的更改*************** */
                    $query1=mysql_query("update tb_admin set admin_pass='".$this->admin_new_pass."'");
                    if($query1==true){
                        echo "<script>alert('密码更改成功');history.back();</script>";
                    }
                }
        }
        }
        /* **********************************类的实例化过程********************************** */
❹     $chk=new
chkuser($_POST[admin_user],md5($_POST[admin_pass]),md5($_POST[admin_new_pass]),$_POST[admin_
new_pass2],$_POST[yzm]);
        $chk->chkinput();
?>
```

代码贴士

❶ chkuser：创建一个类名称为 chkuser，用于修改管理员密码。
❷ chkinput ()：创建一个自定义函数 chkinput()，用于实现密码修改。
❸ strval ()：获取变量的字符串。
❹ new chkuser：实现类的实例化，应用类实现密码修改。

5.9.3 客户信息管理的实现过程

客户信息管理模块使用的数据表：tb_customer

客户信息管理模块的主要功能就是向数据库中添加客户信息，并且可以对指定的客户进行删除操作。客户信息管理模块的运行结果如图 5.37 所示。



图 5.37　客户信息管理模块的运行结果

客户信息管理模块主要通过 customer.php、customer_ok.php 和 delete_customer.php 文件来完成。首先通过 customer.php 文件来添加客户信息，并且输出数据库中的所有客户信息，设置删除客户信息的超链接。其输出客户详细信息的关键代码如下：

例程 11　代码位置：光盘\TM\05\wlgl\customer.php



```
<?php include("conn/conn.php");                                //连接数据库
     $query=mysql_query("select * from tb_customer ");         //执行查询语句，从数据库中读取数据
     while($myrow=mysql_fetch_array($query)){                  //通过 while 循环语句，循环输出数据库中的数据
?>
<tr>
    <td height="26" align="center" bgcolor="#FFFFFF"><?php echo $myrow[customer_user];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $myrow[customer_tel];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $myrow[customer_address];?></td>
     <!-- --------------设置删除客户信息的超链接，并且设置客户的 ID 为变量-------------------- -->
    <td align="center" bgcolor="#FFFFFF"><a href="delete_customer.php?delete=<?php echo
$myrow[customer_id];?>">删除</a></td>
</tr>
  <?php }?>
```

客户信息添加的操作通过 customer_ok.php 文件完成，首先连接数据库，然后判断提交的电话号码格式是否正确，最后将客户信息数据提交到数据库中。程序的代码如下：

例程12　代码位置：光盘\TM\05\wlgl\customer_ok.php

```
<?php include("conn/conn.php");                                //连接数据库
     if($Submit==true and $customer_user==true){               //判断提交的值是否为真
          /* ***************************判断提交的电话号码格式是否正确*************************** */

     if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$customer_tel,$countes)){
               /* ***************************将提交的数据添加到客户信息表中************************ */
               $query=mysql_query("insert into tb_customer(customer_user,customer_tel, customer_address)
values('$customer_user','$customer_tel','$customer_address')");
               if($query==true){                               //如果添加操作成功，弹出提示，并返回首页
                    echo "<script> alert('客户信息添加成功');window.location.href='indexs.php?lmbs=客户信息
管理';</script>";
               }
          }else{                                               //如果添加操作失败，弹出提示，并返回上一页
          echo "<script>alert('您输入的电话号码格式不正确!!');history.back()</script>";
    }
}
?>
```

客户信息删除的操作通过 delete_customer.php 文件来完成，主要以超链接中提交的变量为根据，删除数据库中指定的客户信息。程序的关键代码如下：

例程13　代码位置：光盘\TM\05\wlgl\delete_customer.php

```
<?php include("conn/conn.php");                                //连接数据库
     if($delete==true){                                        //判断提交的变量值是否为真
          /* ********************根据$_GET 获取的变量值为依据，执行删除语句******************** */
          $query=mysql_query("delete from tb_customer   where customer_id='$_GET[delete]'");
          if($query){                                          //如果删除操作成功，则弹出提示，并返回首页
               echo  "<script>  alert('客户信息添加成功');window.location.href='indexs.php?lmbs=客户信息管理
';</script>";
          }
     }
?>
```

5.9.4 车源信息管理的实现过程

车源信息管理模块使用的数据表：tb_car

车源信息管理模块主要实现对车辆信息的添加、修改和删除操作。车源信息管理模块的运行结果如图 5.38 所示。

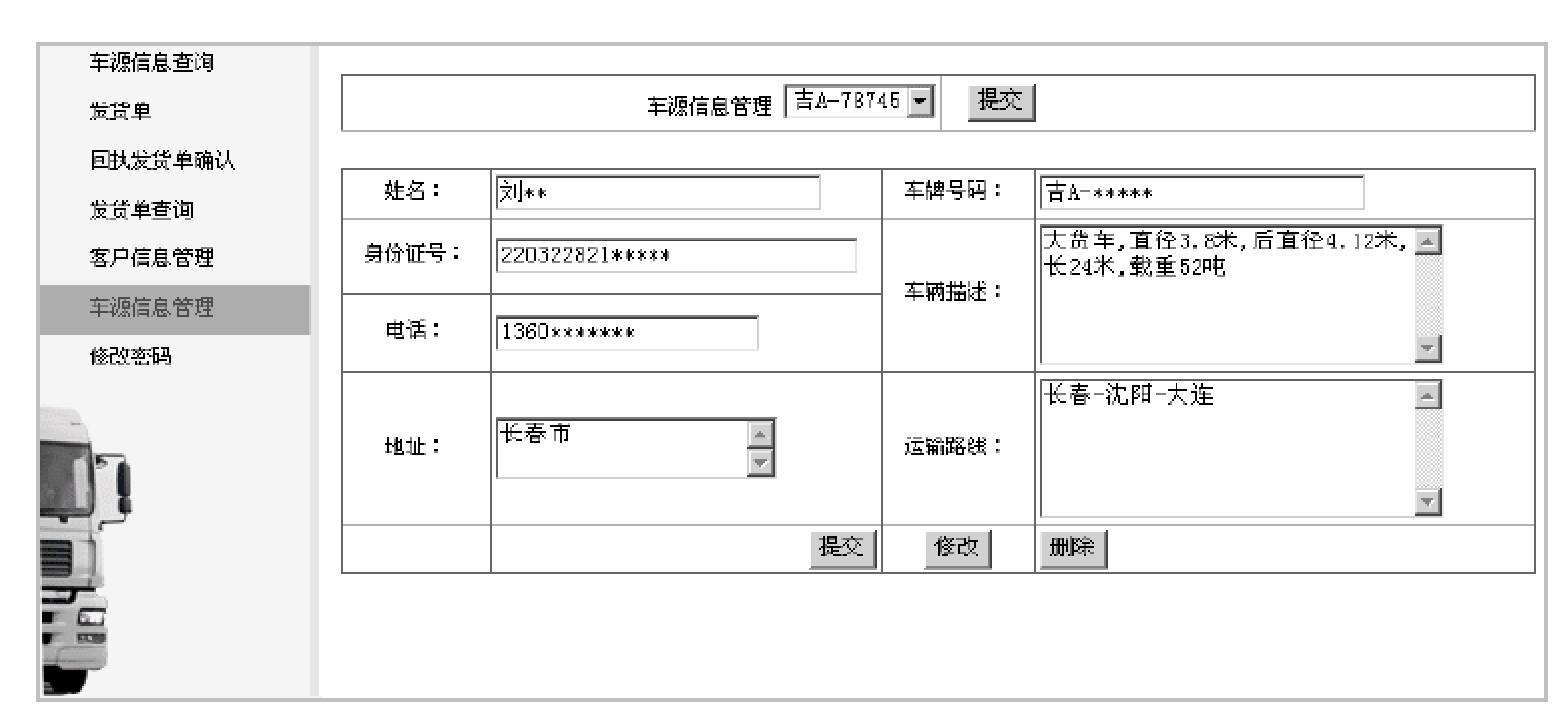


图 5.38 车源信息管理模块的运行结果

车源信息管理模块主要由两个文件组成。一个是 car.php 文件，以表单文件为主，用于输出车辆的信息和提交车辆的信息。设置 form2 表单，根据车牌号码从数据库中读取对应车辆的信息，进而实现对该车辆信息的修改或者删除操作；设置 form1 表单，用于显示从数据库中读取的车辆信息和直接添加车辆信息到数据库。form1 表单中使用的重要元素如表 5.4 所示。

表 5.4 车源信息管理模块中使用的重要表单元素

名 称	元 素 类 型	重 要 属 性	含 义
form1	form	method="post" action="car_insert_ok.php" onSubmit="javascript:return check_car(form1);"	表单
username	text	id="username" value="<?php echo $myrows[username];?>"	车主姓名
car_number	text	id="car_number" value="<?php echo $myrows[car_number];?>"	车牌号码
car_tel	text	value="<?php echo $myrow[tel];?>"	车主电话
user_number	text	id="user_number" value="<?php echo $myrows[user_number];?>"	身份证号码
car_content	textarea	<?php echo $myrows[car_content];?>	车辆描述
address	textarea	<?php echo $myrows[address];?>	家庭地址
car_road	textarea	<?php echo $myrows[car_road];?>	运输路线
Submit	submit	value="提交"	提交表单
Submit2	submit	value="修改"	修改表单
Submit4	submit	value="删除"	删除表单

另一个是 car_insert_ok.php 文件，通过该文件实现对表单中提交的数据进行处理。首先连接数据库，然后根据表单中提交的值进行判断，当按钮$Submit 的值为“提交”时，将提交的数据存储到指定的数据表中；当按钮$Submit2 的值为“修改”时，则更新语句，对车辆的信息进行更新；当按钮$Submit4 的值为“删除”时，则执行删除语句，将指定的车辆信息删除。程序关键代码如下：



例程14　代码位置：光盘\TM\05\wlgl\car_insert_ok.php

```
<?php session_start(); include("conn/conn.php");                    //初始化 session 变量，连接数据库
❶  if($_SESSION[admin_user]==true){                                   //判断管理员是否是正常登录
       if($Submit=="提交"){                                            //判断表单提交的值是否为“提交”
           /* ********************判断表单中提交的电话号码格式是否正确******************** */
❷      if(preg_match("/\d{17}[\d|X]|\d{15}/",$user_number,$counts)){
         if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$user_tel,$countes)){
           /* ************************将表单中的数据添加到 tb_car 数据表中******************** */
           $query=mysql_query("insert into tb_car(username,user_number,tel,address,car_number, car_road,
car_ content) values('$username','$user_number','$user_tel','$address','$car_number','$car_road','$car_content')");
               if($query==true){
                   echo "<script>alert('车源信息添加成功!');window.location.href='indexs.php?lmbs=车
源信息管理';</script>"; }
           }else{
               echo "<script>alert('您输入的电话号码格式不正确!!');history.back()</script>";    }
           }else{
               echo "<script>alert('您输入的身份证号码的格式不正确!!');history.back()</script>";     }
       }
       /* *****************************判断表单提交的值是否为“修改” ************************* */
❸      if($Submit2=="修改"){
           /* **************************判断电话号码的格式是否正确************************** */
           if(preg_match("/\d{17}[\d|X]|\d{15}/",$user_number,$counts)){
           if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$user_tel,$countes)){
           /* ****************************更新 tb_car 数据表中的数据*************************** */
           $query="update tb_car set username='$username', user_number='$user_number', tel='$user_tel',
address='$address', car_number='$car_number', car_road='$car_road', car_content='$car_content'";
           $result=mysql_query($query);
               if($result==true){
                   echo "<script>alert('车源数据更新成功!!');window.location.href='indexs.php?lmbs=车
源信息管理';</script>";}
           }else{
               echo "<script>alert('您输入的电话号码格式不正确!!');history.back()</script>";}
           }else{
               echo "<script>alert('您输入的身份证号码的格式不正确!!');history.back()</script>";   }
       }
       /* ****************************判断表单提交的值是否为“删除” **************************** */
❹      if($Submit4=="删除"){
           $query="delete   from tb_car where car_number='$car_number'";  //删除 tb_car 数据表中的数据
           $result=mysql_query($query);
               if($result==true){
                   echo "<script>alert('车源数据删除成功!');window.location.href='indexs.php?lmbs=车
源信息管理';</script>";
               }
       }
?>
<?php   }else{
echo "<script>alert('请您正确登录!'); window.location.href='index.php';</script>";
}
?>
```

代码贴士

❶ $_SESSION[admin_user]：判断用户是否是正常登录。

❷ preg_match()：应用 preg_match 验证电话号码的格式是否正确。preg_match()对正则表达式进行匹配。返回匹配的次数，0 次（没有匹配）或者 1 次，因为 preg_match()函数在第一次匹配之后将停止搜索，出错返回 false。

❸ $Submit2=="修改"：判断表单提交的值是否等于“修改”，如果等于“修改”则执行修改的操作。

❹ $Submit4=="删除"：判断表单提交的值是否等于“删除”，如果等于“删除”则执行删除语句。

5.10 开发技巧与难点分析

5.10.1 应用存储过程实现管理员登录

在物流配送信息网系统中，管理员登录是通过存储过程来实现的。下面就来讲解一下如何创建存储过程和应用存储过程。

MySQL 5.0 以后的版本中开始支持存储过程，存储过程具有一致性、高效性、安全性和体系结构等特点，其将在以后的项目中得到广泛的应用。

一个完整的存储过程包括名称、参数列表，以及很多 SQL 语句集。下面以一个存储过程的定义过程为例，对其结构进行详细的讲解。

```
❶ create procedure procedure_name (in parameter integer)
❷ begin
declare variable varchar(20);
if parameter=1 then
set variable='Apache';
else
set variable='MySQL';
end if;
insert into tb_table (tb_name) values (variable);
❸ end;
```

代码贴士

❶ create procedure：创建存储过程，以关键字 create procedure 开始，后面紧跟存储过程的名称和参数。

❷ begin：MySQL 存储过程的语句块以 begin 开始。

❸ end：MySQL 存储过程的语句块以 end 结束。

MySQL 中存储过程的建立以关键字 create procedure 开始，后面紧跟存储过程的名称和参数。MySQL 的存储过程名称不区分大小写，例如 PROCEDURES()和 procedures()代表同一存储过程名。存储过程名不能与 MySQL 数据库中的内建函数重名。

存储过程的参数一般由 3 部分组成。第一部分可以是 in、out 或 inout。in 表示向存储过程中传入参数；out 表示向外传出参数；inout 表示定义的参数可传入存储过程，并可以被存储过程修改后传出存储过程。存储过程默认为传入参数，所以参数 in 可以省略。第二部分为参数名。第三部分为参数的类型，该类型为 MySQL 数据库中所有可用的字段类型，如果有多个参数，参数之间可以用逗号进行分割。



MySQL 存储过程的语句块以 begin 开始，以 end 结束。语句体中可以包含变量的声明、控制语句、SQL 查询语句等。由于存储过程内部语句要以分号结束，所以在定义存储过程前应将语句结束标志“;”更改为其他字符，并且该字符在存储过程中出现的几率也应该较低，可以用关键字 delimiter 更改，例如：

```
mysql>delimiter //
```

注意　在应用 delimiter 关键字更改存储过程中的语句结束标志“;”时，所使用的字符必须与 delimiter 关键字之间保留一个空格，否则该存储过程将会出现错误，不能成功创建。

存储过程创建之后，可用如下语句进行删除，参数 procedure_name 指存储过程名。

```
drop procedure procedure_name
```

MySQL 存储过程是在命令提示符下创建的，其操作步骤如下：

（1）打开命令提示符。

（2）进入命令提示符后，首先登录 MySQL 数据库服务器，在命令提示符下输入如下命令：

```
mysql –u 用户名 –p 用户密码
```

（3）更改语句结束符号，将语句结束符更改为“//”，代码如下：

```
delimiter //
```

（4）创建存储过程前应首先选择某个数据库，代码如下：

```
use 数据库名
```

（5）创建存储过程。

（6）通过 call 语句调用存储过程。

下面以本项目中的管理员登录模块为例，对存储过程进行实例化的讲解。首先创建存储过程，在命令提示符中输入代码，如图 5.39 所示。

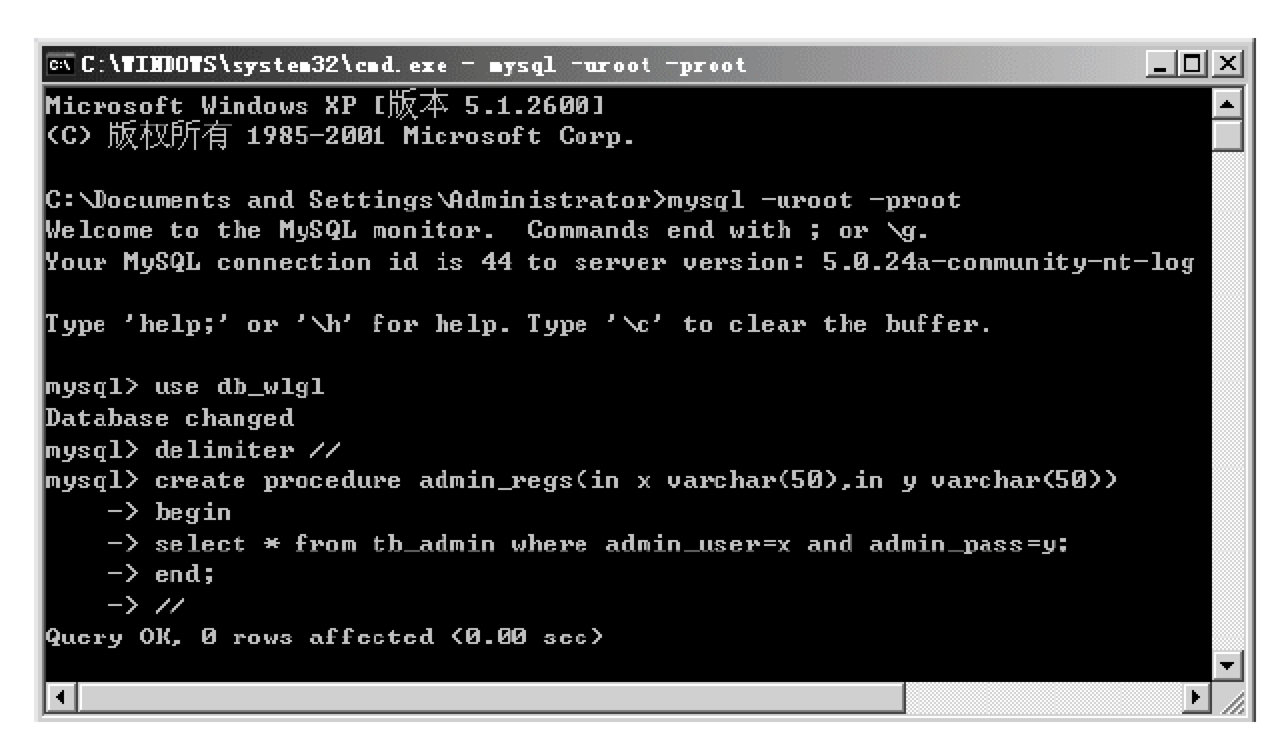


图 5.39　创建存储过程

然后在 index.php 文件中编写 PHP 脚本语句，调用存储过程实现用户登录。关键代码如下：

例程15 代码位置：光盘\TM\05\wlgl\index.php

```
<?php
    if($_POST[admin_user]!="" || $_POST[admin_pass]!=""){                    //获取表单提交的数据
        $conn=new mysqli("localhost","root","root","db_wlgl");               //连接数据库
        $conn->query("set names gb2312");                                     //指定字符编码格式
        $admin_user=$_POST[admin_user];                                       //获取变量值
        $admin_pass=md5($_POST[admin_pass]);                                  //获取变量值
        $sql=$conn->query("call admin_reg('".$admin_user."','".$admin_pass."')");//调用存储过程
        $res=$sql->fetch_array(MYSQL_BOTH);                                   //输出结果
        if($res!=NULL){
            session_register("admin_user");                                   //定义 session 变量
            session_register("admin_pass");
            echo "<script>alert('用户注册成功!');window.location.href='indexs.php';</script>";
        }else{
        echo "<script>alert('用户注册失败!');</script>";
        }
    }
?>
```

5.10.2 应用正则表达式验证电话号码

在物流配送信息网中，确认输入的电话号码正确与否是一个非常重要的问题，如果客户在填写订单时使用了错误的电话号码，那么就会带来很多不必要的麻烦，所以在填写电话号码时一定要注意电话号码的准确性。为了更好地确保电话号码输入的准确性，可以通过程序对其进行控制。虽然程序不能判断电话号码输入得是否正确，但是通过程序可以对电话号码的格式进行判断，从而避免一些电话号码在录入的过程中出现多写或者漏写数字的错误。

验证电话号码格式是否正确可以通过正则表达式来完成。通过正则表达式可以有效地判断输入电话号码的格式是否正确。

在进行判断之前首先要对电话号码的格式进行分类：第一类区号是 3 位数字，其他是 8 位数字；第 2 类区号是 4 位数字，其他是 7 位数字；第 3 类区号是 4 位数字，其他是 8 位数字；第 4 类是手机电话号码，由 11 位数字组成。

然后编写一个判断电话号码格式的正则表达式，其代码如下：

```
/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/
```

正则表达式的功能分析如下：使用“^”和“$”对字符串进行边界限制，对区号从字符串的开始进行匹配，对其他号码从字符串的末尾开始进行匹配；将括号“()”中的内容作为一个原子使用；使用“\d”来匹配一个数字，区号为 3 或 4 个数字，其他数字为 7 或 8 个，或者直接判断为 11 个数字；使用“{}”来对前字符进行重复匹配；使用“|”对匹配的模式进行选择，分成 4 个模式。

正则表达式创建完成后就可以应用到程序中对电话号码进行判断，要完成对电话号码的判断还需要应用 preg_match()函数。

preg_match()函数对正则表达式进行匹配。返回匹配的次数，0 次（没有匹配）或者 1 次，如果出错则返回 false。preg_match()函数的参数说明如表 5.5 所示。





表 5.5　preg_match()函数的参数说明

参　数	说　明
pattern	必要参数，指定匹配的正则表达式
subject	必要参数，指定要搜索的字符串
subpatterns	可选参数，如果提供了 subpatterns，则其会被搜索的结果所填充。$ subpatterns [0] 将包含与整个模式匹配的文本，$ subpatterns [1] 将包含与第一个捕获的括号中的子模式所匹配的文本，以此类推
flags	可选参数，如果设定标记 PREG_OFFSET_CAPTURE，则每个出现的匹配结果也同时返回其附属的字符串偏移量。注意这改变了返回数组的值，使其中的每个单元也是一个数组，其中第一项为匹配字符串，第二项为其偏移量。本参数自 PHP 4.3.0 开始使用

最后将正则表达式和 preg_match()函数应用到实际的程序中，实现对电话号码格式的判断。这里以 car_insert_ok.php 文件中的代码为例，关键代码如下：

例程16　代码位置：光盘\TM\05\wlgl\car_insert_ok.php

```
<?php
if($Submit==true){                                                    //判断表单提交的值
      /* ***************应用 preg_match()函数和正则表达式判断身份证号码格式是否正确*************** */
      if(preg_match("/\d{17}[\d|X]|\d{15}/",$user_number,$counts)){
     /* **************应用 preg_match()函数和正则表达式判断电话号码格式是否正确****************** */
      if(preg_match("/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$|^(\d{11})$/",$user_tel,$countes)){
            /* **********如果格式正确，则执行下面的插入语句，将车辆信息添加到数据库中********** */
            $query=mysql_query("insert into tb_car(username,user_number,tel,address,car_number,car_road, car_
content) values('$username','$user_number','$user_tel','$address','$car_number','$car_road','$car_content')");
            if($query==true){
                  echo "<script>alert('车源信息添加成功!');window.location.href='indexs.php?lmbs=车源信息管理';
</script>";      }
       }else{
            echo "<script>alert('您输入的电话号码格式不正确!!');history.back()</script>";      }
      }else{
                  /* ***********************否则弹出提示信息，返回到表单提交页*********************** */
               echo "<script>alert('您输入的身份证号码的格式不正确!!');history.back()</script>"; }
}
?>
```

5.11　报表打印技术

报表打印技术的实现方法很多，可以通过 Web 打印、Word 打印、Excel 打印和 CSS 样式打印等。在本项目中应用的打印方式是通过 Web 打印，并且是打印指定框架中的内容，其详细的讲解可参考 5.7.5 节。下面再介绍一种通过 CSS 样式来打印页面中指定内容的方法。

通过 CSS 样式打印页面中指定内容主要会用 CSS 样式的 media 类型。media 类型是 CSS 属性媒体类型，用于直接引入媒体的属性。语法格式如下：

```
@media screen | print | projection | braille | aural | tv | handheld | all
```

media 类型的参数说明如表 5.6 所示。

表 5.6 media 类型的参数说明

参　数	说　明
screen	默认值，指提交到计算机屏幕
print	指输出到打印机
projection	指提交到投影机
braille	提交到凸字触觉感知设备
aural	指语音电子合成器
tv	电视类型的媒体
handheld	指手持式显示设备
all	用于所有媒体

注意 应用 CSS 样式打印，控制的是指定表格的背景，只是将表格的背景颜色进行隐藏，而不能控制表格中内容的输出。所以在应用 CSS 样式进行指定内容的打印时，前提是不打印的内容必须是以表格中背景图像的形式显示的，否则应用 CSS 样式进行指定内容打印是没有效果的。

这里以项目中的客户信息为例，创建一个 css_print.php 文件，讲解通过 CSS 样式打印指定内容的操作过程。首先编写用于控制指定内容的不打印 CSS 样式，程序代码如下：

```
<style>
@media print{
div{display:none}
.bgnoprint{
	background:display:none;
}
.noprint{
	display:none
}
}
body {
	margin-top: 0px;
}
</style>
```

然后设计页面的布局，将页面的头文件和尾文件以插入背景图片的方式进行实现，连接数据库，应用 while 循环语句输出数据库中的客户信息，并且指定页面的头文件和尾文件为不打印内容。

```
<body class="bgnoprint">
<!-- ----------------------------------------------------应用 bgnoprint 样式----------------------------------------------- -->
<table width="935" height="145" border="0" cellpadding="0" cellspacing="0">
  <tr>
     <!--设置头文件，以背景图像的形式进行输出，并且应用 bgnoprint 样式，控制该背景图像为不打印的内容-->
    <td background="images/index_03.gif" class="bgnoprint"> </td>
  </tr>
</table>
<!-- -----------------------------------------连接数据库，循环输出数据库中数据--------------------------------------- -->
 <?php
 include("conn/conn.php");                                        //连接数据库
```



```
 $query=mysql_query("select * from tb_customer ");          //执行查询语句
 while($myrow=mysql_fetch_array($query)){                    //循环输出数据库中的内容
  ?>
  <tr>
    <td height="26" align="center" bgcolor="#FFFFFF"><?php echo $myrow[customer_user];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $myrow[customer_tel];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $myrow[customer_address];?></td>
    <td align="center" bgcolor="#FFFFFF"><a href="delete_customer.php?delete=<?php echo
$myrow[customer_id];?>">删除</a></td>
  </tr>
<?php }?>
</table>
```

最后设置一个 div 标签，建立超链接，应用 onClick 事件调用 window.print()方法和 printview()函数实现页面的打印和打印预览的功能。

```
<!--设置超链接，应用 onClick 事件，调用 window.print()方法执行打印，调用 printview()函数实现打印预览的功能-->
<tr>
    <td height="26" colspan="4" align="center" bgcolor="#FFFFFF"><div><a href="#" onClick= "window.print();"> 客户详单打印</a> <a href="#" onClick="printview();">打印预览</a></div></td>
</tr>
```

通过 CSS 样式实现页面中指定内容打印的运行结果如图 5.40 和图 5.41 所示。

图 5.40　客户信息打印页面的运行结果



图 5.41　客户信息打印预览页面的运行结果

技巧　在进行 Web 打印时，可以通过以下操作控制是否打印背景颜色和图像。在 IE 窗口中，选择“工具”/“Internet 选项”命令，在弹出的“Internet 选项”对话框中选择“高级”选项卡，在“设置”列表中设置“打印背景颜色和图像”前面的复选框是否选中，如果选中，代表打印背景颜色和图像，否则不打印背景颜色和图像。

5.12　本 章 总 结

本章从某物流配送公司的实际需求角度出发，开发一个完整的物流配送信息管理系统，详细地讲解了物流配送信息网的开发流程，从最初的需求分析、可行性分析，到系统的设计、数据库的设计，其中重点突出车源信息查询模块、发货单管理模块和回执单验收管理模块的设计。通过本章的学习，读者不但可以了解物流配送系统开发的整体思路，而且能够掌握很多关键的技术和技巧：函数的灵活运用、MySQL 存储过程的创建和应用、报表打印技术的实现和应用正则表达式验证电话号码等。

第 6 章

学校图书馆管理系统

（Apache+PHP+phpMyAdmin+MySQL 5.0 实现）

随着网络技术的高速发展和计算机应用的普及，利用计算机对图书馆的日常工作进行管理势在必行。虽然目前很多大型的图书馆已经有一整套比较完善的管理系统，但是在一些中小型的图书馆中，大部分工作仍需由手工完成，工作起来效率比较低，管理员不能及时了解图书馆内各类图书的借阅情况，读者需要的图书难以在短时间内找到，不便于动态及时地调整图书结构。为了更好地适应当前读者的借阅需求，解决手工管理中存在的许多弊端，越来越多的中小型图书馆正在逐步向计算机信息化管理转变。本章通过开发一个流行的图书馆管理系统，为读者讲解详细的项目开发流程。通过阅读本章，可以学习到：

- 图书馆管理系统开发的基本过程
- 系统设计的方法
- 如何分析并设计数据库、数据表
- 多表查询的方法
- 面向对象的编程方法
- 主要功能模块的实现方法
- 单元测试的方法
- 如何自动计算图书归还日期
- 内联接和外联接语句的使用方法



6.1　开发背景

博考图书馆是一家吉林××师范学校的大型图书馆。随着学校图书馆规模的不断壮大，经营的图书品种、数量也逐渐增多。在学校图书馆不断发展的同时，校图书馆常年采用的传统的人工方式管理暴露了一些问题。例如，查找读者借阅的某一本图书的具体摆放位置，需要靠人工记忆在书海中苦苦查找，由于图书储存量大，很难准确定位图书的具体位置，因此每天都要浪费大量宝贵的时间资源。学校图书馆为提高工作效率，同时摆脱图书管理人员在工作中出现的种种弊端，现需要委托某单位开发一个学校图书馆管理系统。

6.2　需求分析

通过计算机对图书进行管理，不仅为图书馆的管理注入了新的生机，而且在运营过程中还节省了大量的人力、物力、财力和时间；不仅提高了图书馆的工作效率，还为图书馆在读者群中树立了一个全新的形象，为图书馆日后发展奠定了一个良好的基础。通过对一些大型图书馆的实际考察、分析，并结合图书馆的要求以及实际的市场调查，要求本系统具有以下功能：

☑　网站设计页面要求美观大方、个性化，功能全面，操作简单。
☑　要求实现基础信息的管理平台。
☑　要求对所有读者进行管理。
☑　要求实现图书借阅排行、了解当前的畅销书。
☑　商品分类详尽，可按不同类别查看图书信息。
☑　提供快速的图书信息、图书借阅检索功能，保证数据查询的灵活性。
☑　实现图书借阅、图书续借、图书归还的功能。
☑　实现综合条件查询，如按用户指定条件查询、按日期时间段查询、综合条件查询等。
☑　要求图书借阅、续借、归还时记下每一笔记录的操作员。
☑　实现对图书借阅、续借和归还过程的全程数据信息跟踪。
☑　提供借阅到期提醒功能，使管理者可以及时了解到已经到达归还日期的图书借阅信息。
☑　提供灵活、方便的权限设置功能，使整个系统的管理分工明确。
☑　具有易维护性和易操作性。

6.3　系统设计

6.3.1　系统目标

根据前面所作的需求分析及用户的需求可以得出，学校图书馆管理系统实施后，应达到以下目标：

☑　网站设计页面要求美观大方、功能全面，操作简单。
☑　网站整体结构和操作流程合理顺畅，实现人性化设计。

☑ 规范、完善的基础信息设置。
☑ 对操作员设置不同的操作权限，为管理员提供修改权限功能。
☑ 对所有读者进行集中管理。
☑ 对图书信息进行集中管理。
☑ 实现图书借阅排行，以便了解当前的畅销书。
☑ 提供快速的图书信息、图书借阅检索功能。
☑ 实现图书借阅、图书续借、图书归还功能。
☑ 实现综合条件查询，如按用户指定条件查询、按日期时间段查询、综合条件查询等。
☑ 实现图书借阅、续借、归还时记下每一笔记录的操作员。
☑ 支持图书到期提醒功能。
☑ 为操作员提供密码修改功能。
☑ 系统运行稳定、安全可靠。

6.3.2 系统功能结构

根据学校图书馆管理系统的特点，可以将其分为系统设置、读者管理、图书档案管理、图书借还、系统查询等 5 个部分，其中各个部分及其包括的具体功能模块如图 6.1 所示。

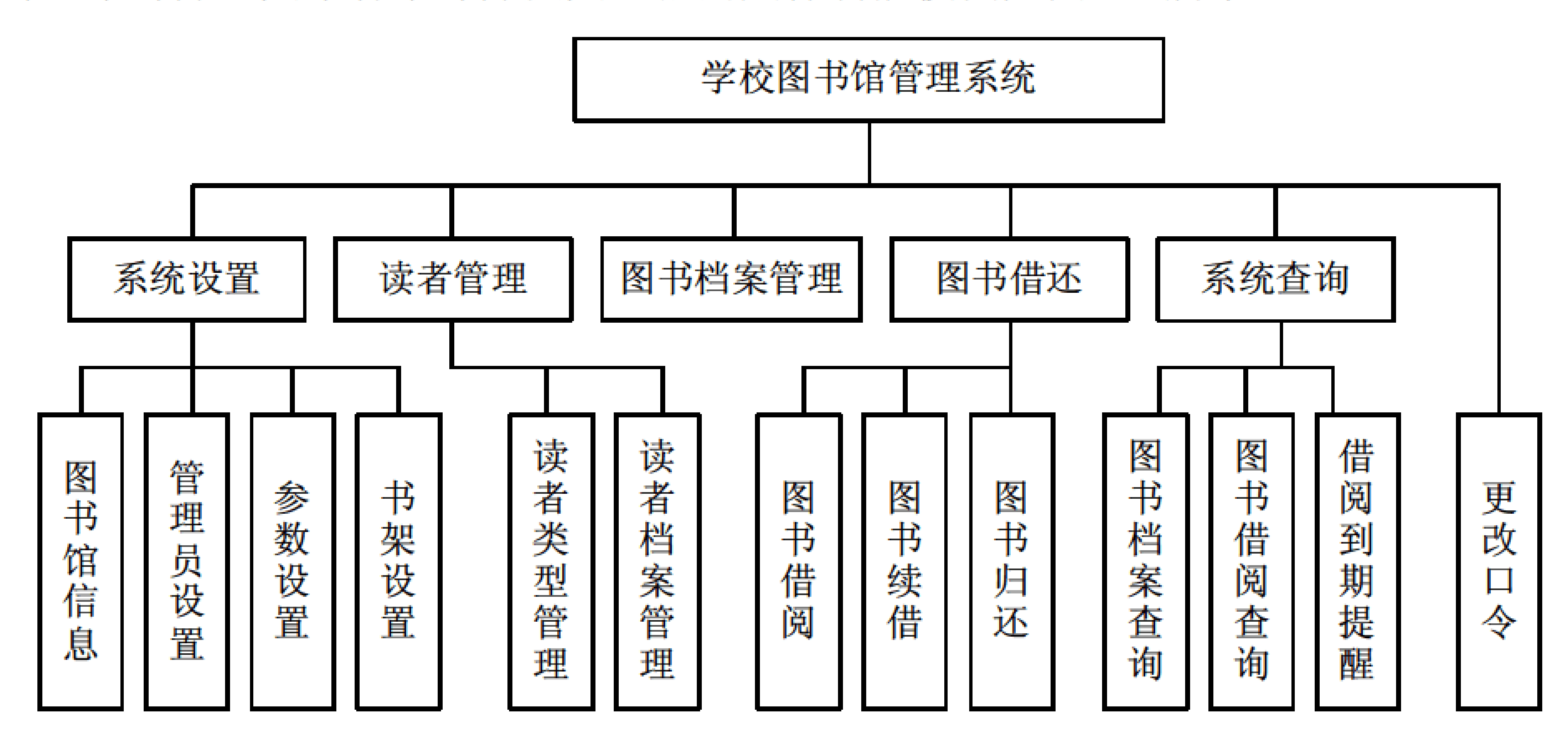


图 6.1　学校图书馆管理系统功能结构图

6.3.3 系统流程图

学校图书馆管理系统的流程如图 6.2 所示。

6.3.4 系统预览

学校图书馆管理系统由多个程序页面组成，下面仅列出几个典型页面，其他页面参见光盘中的源程序。

系统登录页面如图 6.3 所示，该页面用于实现管理员登录。系统首页如图 6.4 所示，该页面用于实现显示系统导航、图书借阅排行和版权信息等功能。





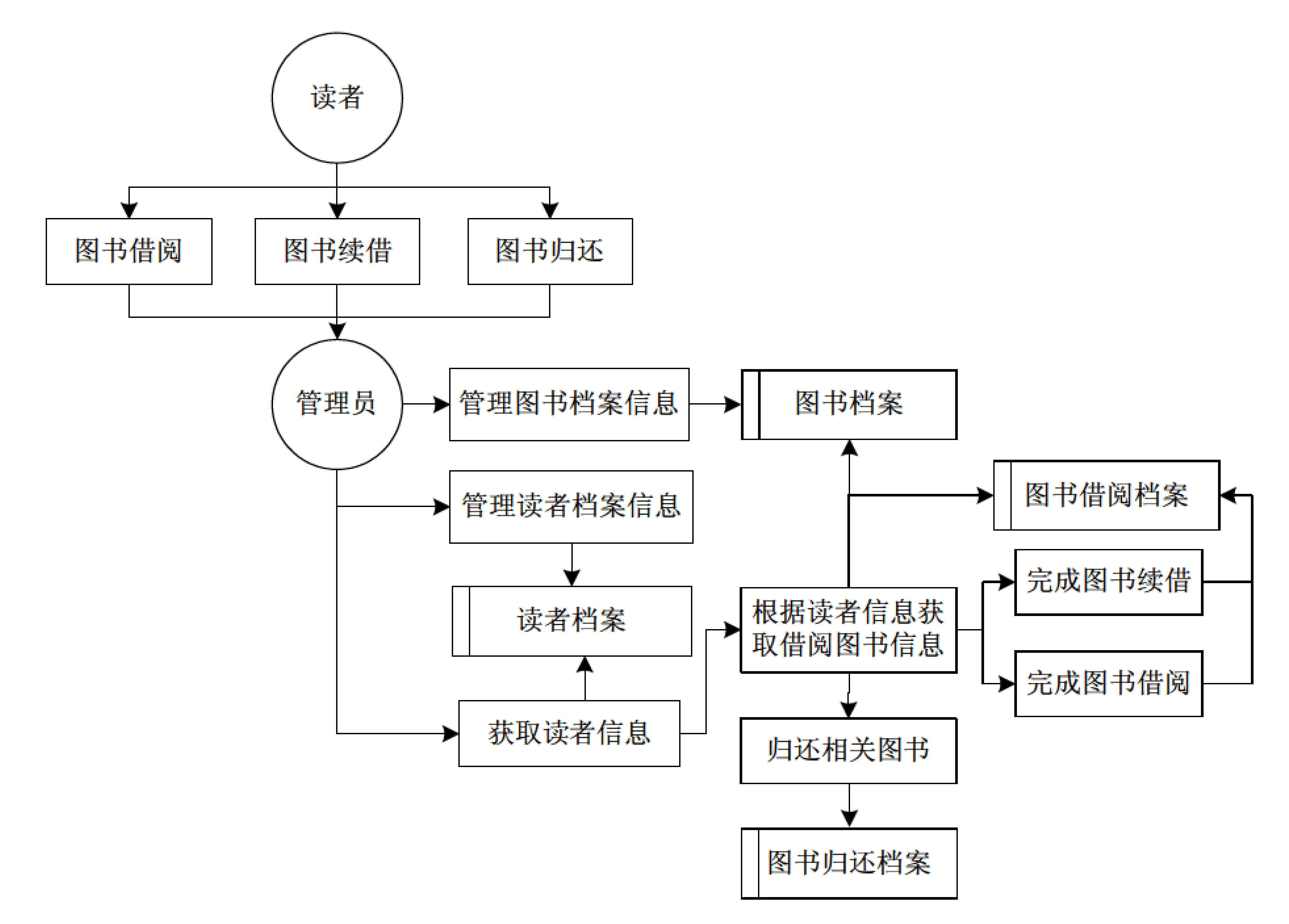


图 6.2　学校图书馆管理系统流程图




图 6.3　系统登录（光盘\TM\06\library\login.php）



图 6.4　系统首页（光盘\TM\06\library\index.php）

图书借阅页面如图 6.5 所示，该页面用于实现图书借阅功能。图书借阅查询页面如图 6.6 所示，该页面用于实现按照复合条件查询图书借阅信息。



图 6.5　图书借阅（光盘\TM\06\library\bookBorrow.php）



图 6.6　图书借阅查询（光盘\TM\06\library\borrowQuery.php）



6.3.5　开发环境

在开发学校图书馆管理系统时，该项目使用的软件开发环境如下。

1．服务器端

- ☑ 操作系统：Windows Server 2003/Linux（推荐）。
- ☑ 服务器：Apache 2.2.8。
- ☑ PHP 软件：PHP 5.2.6。
- ☑ 数据库：MySQL 5.0.51。
- ☑ MySQL 图形化管理软件：phpMyAdmin-2.10.3。
- ☑ 开发工具：Dreamweaver 8。
- ☑ 浏览器：IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

2．客户端

- ☑ 浏览器：IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

6.3.6　文件夹组织结构

在编写代码之前，可以把系统中可能用到的文件夹先创建出来（例如，创建一个名为 Images 的文件夹，用于保存网站中所使用的图片），这样不但可以方便以后的开发工作，也可以规范网站的整体架构。笔者在开发学校图书馆管理系统时，设计了如图 6.7 所示的文件夹组织结构图。在开发时，只需要将所创建的文件保存在相应的文件夹中即可。

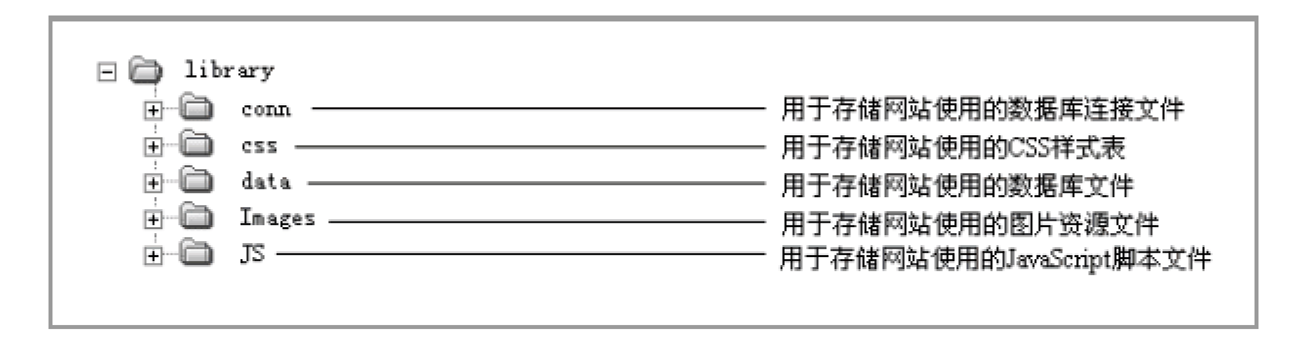


图 6.7　文件夹组织结构

6.4　数据库设计

学校图书馆管理系统是一个数据库开发的 Web 网站。下面对学校图书馆使用的数据库进行分析和介绍。

6.4.1　数据库分析

由于本系统是为中小型的图书馆开发的程序，需要充分考虑到成本问题及使用需求（如跨平台）等问题，而 MySQL 是世界上最为流行的开放源码的数据库，是完全网络化的跨平台的关系型数据库





系统，这正好满足了中小型企业的需求，所以本系统采用 MySQL 数据库。

6.4.2　数据库概念设计

根据以上各节对系统所做的需求分析、系统设计，规划出本系统中使用的数据库实体分别为图书档案实体、读者档案实体、借阅档案实体、归还档案实体和管理员实体。下面将介绍几个关键实体的 E-R 图。

1．图书档案实体

图书档案实体包括编号、条形码、书名、类型、作者、译者、出版社、价格、页码、书架、录入时间和操作员等属性。图书档案实体的 E-R 图如图 6.8 所示。

2．读者档案实体

读者档案实体包括编号、姓名、性别、条形码、职业、出生日期、有效证件、证件号码、电话、电子邮件、登记日期、操作员、类型和备注等属性。读者档案实体的 E-R 图如图 6.9 所示。

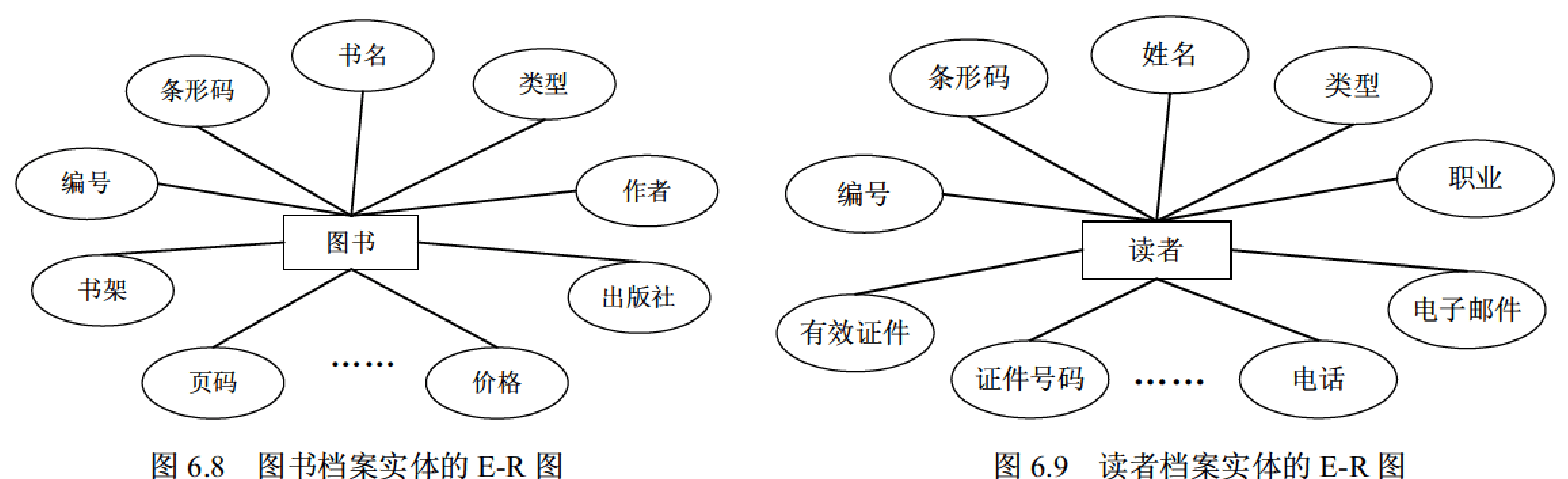


图 6.8　图书档案实体的 E-R 图　　图 6.9　读者档案实体的 E-R 图

3．借阅档案实体

借阅档案实体包括编号、读者编号、图书编号、借书时间、应还时间、操作员和是否归还等属性。借阅档案实体的 E-R 图如图 6.10 所示。

4．归还档案实体

归还档案实体包括编号、读者编号、图书编号、归还时间和操作员等属性。归还档案实体的 E-R 图如图 6.11 所示。

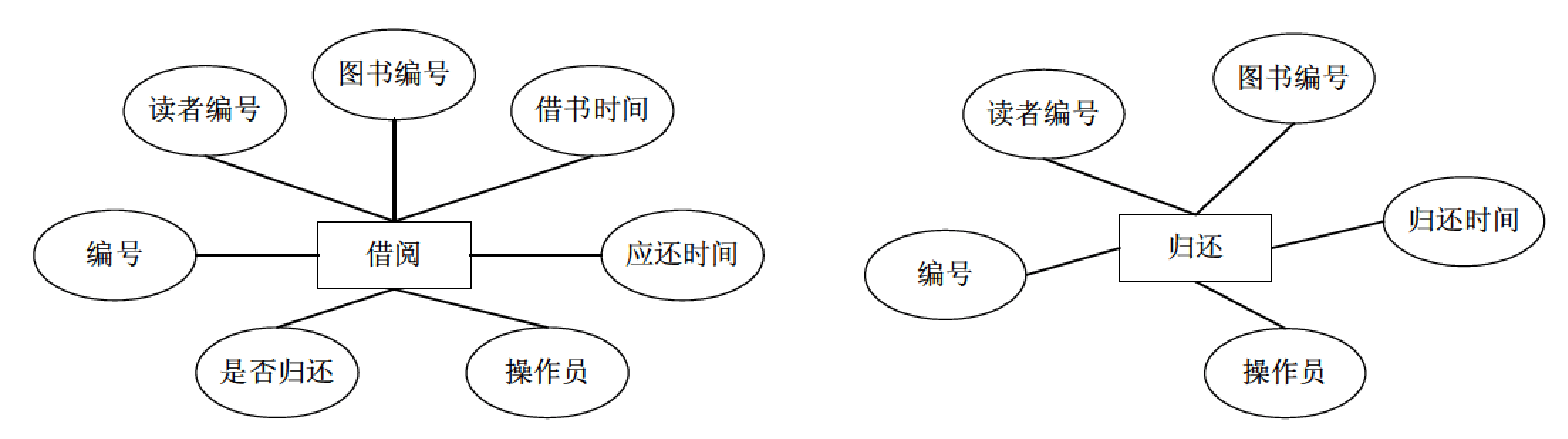


图 6.10　借阅档案实体的 E-R 图　　图 6.11　归还档案实体的 E-R 图

6.4.3　创建数据库及数据表

结合实际情况及对用户需求的分析，学校图书馆管理系统 db_library 数据库主要包含 10 个数据表，如图 6.12 所示。

结合数据表的创建方法，读者可以自行创建以下数据表。

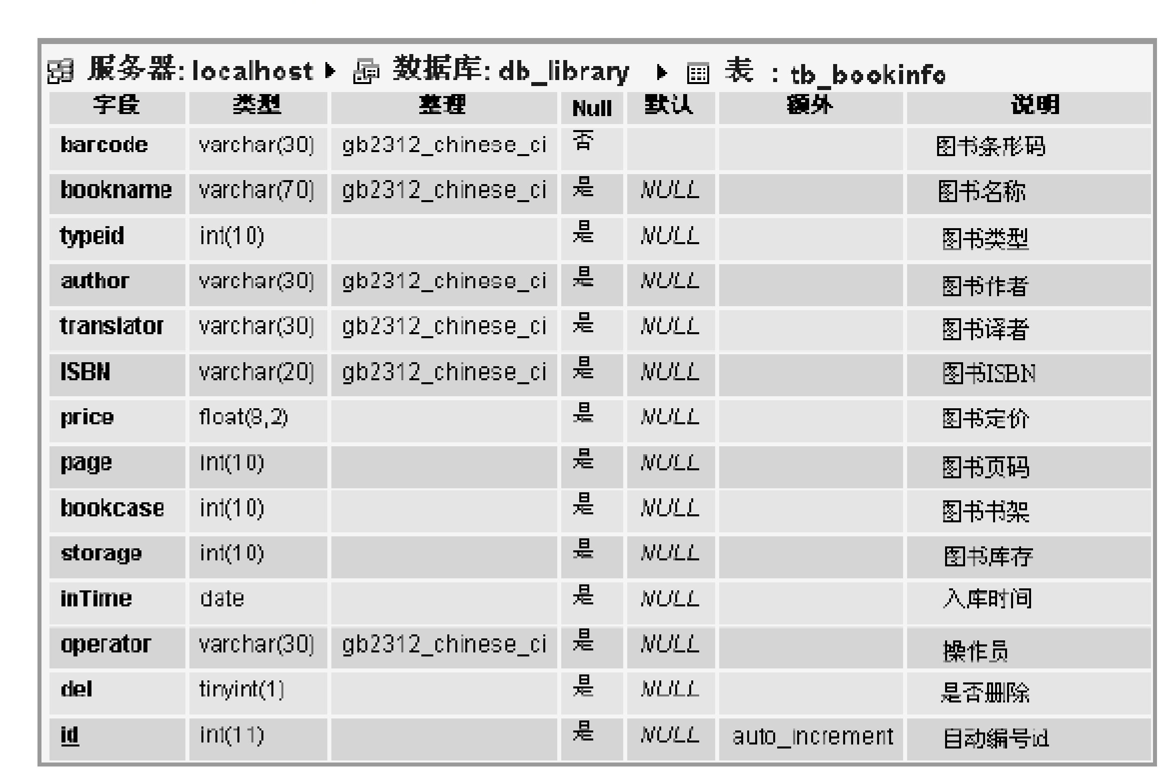
1．tb_bookinfo（图书信息表）

图书信息表主要用于存储图书的基础信息。该数据表的结构如图 6.13 所示。

服务器: localhost ▸ 数据库: db_library

表	类型	整理	说明
tb_bookcase	MyISAM	gb2312_chinese_ci	图书书架信息表
tb_bookinfo	MyISAM	gb2312_chinese_ci	图书信息表
tb_borrow	MyISAM	gb2312_chinese_ci	图书借阅信息表
tb_library	MyISAM	gb2312_chinese_ci	图书馆信息表
tb_manager	MyISAM	gb2312_chinese_ci	管理员信息表
tb_parameter	MyISAM	gb2312_chinese_ci	参数设置信息表
tb_publishing	MyISAM	gb2312_chinese_ci	出版社信息表
tb_purview	MyISAM	gb2312_chinese_ci	权限信息表
tb_reader	MyISAM	gb2312_chinese_ci	读者信息表
tb_readertype	MyISAM	gb2312_chinese_ci	读者类型信息表

图 6.12　学校图书馆管理系统数据表



服务器: localhost ▸ 数据库: db_library ▸ 表 : tb_bookinfo

字段	类型	整理	Null	默认	额外	说明
barcode	varchar(30)	gb2312_chinese_ci	否			图书条形码
bookname	varchar(70)	gb2312_chinese_ci	是	NULL		图书名称
typeid	int(10)		是	NULL		图书类型
author	varchar(30)	gb2312_chinese_ci	是	NULL		图书作者
translator	varchar(30)	gb2312_chinese_ci	是	NULL		图书译者
ISBN	varchar(20)	gb2312_chinese_ci	是	NULL		图书ISBN
price	float(8,2)		是	NULL		图书定价
page	int(10)		是	NULL		图书页码
bookcase	int(10)		是	NULL		图书书架
storage	int(10)		是	NULL		图书库存
inTime	date		是	NULL		入库时间
operator	varchar(30)	gb2312_chinese_ci	是	NULL		操作员
del	tinyint(1)		是	NULL		是否删除
id	int(11)		是	NULL	auto_increment	自动编号id

图 6.13　图书信息表结构

2．tb_borrow（图书借阅信息表）

图书借阅信息表主要用于存储图书的借阅信息。该数据表的结构如图 6.14 所示。

3．tb_reader（读者信息表）

读者信息表主要用于存储读者的基础信息。该数据表的结构如图 6.15 所示。

服务器: localhost ▸ 数据库: db_library ▸ 表 : tb_borrow

字段	类型	整理	Null	默认	额外	说明
id	int(10)		否		auto_increment	自动编号id
readerid	int(10)		是	NULL		读者id编号
bookid	int(10)		是	NULL		图书id编号
borrowTime	date		是	NULL		图书借阅时间
backTime	date		是	NULL		图书归还时间
operator	varchar(30)	gb2312_chinese_ci	是	NULL		操作员
ifback	tinyint(1)		是	0		是否归还

图 6.14　图书借阅信息表结构



服务器: localhost ▸ 数据库: db_library ▸ 表 : tb_reader

字段	类型	整理	Null	默认	额外	说明
id	int(10)		否		auto_increment	自动编号id
name	varchar(20)	gb2312_chinese_ci	是	NULL		读者姓名
sex	varchar(4)	gb2312_chinese_ci	是	NULL		性别
barcode	varchar(30)	gb2312_chinese_ci	是	NULL		读者条形码
vocation	varchar(50)	gb2312_chinese_ci	是	NULL		读者职业
birthday	date		是	NULL		出生日期
paperType	varchar(10)	gb2312_chinese_ci	是	NULL		读者证件
paperNO	varchar(20)	gb2312_chinese_ci	是	NULL		证件号码
tel	varchar(20)	gb2312_chinese_ci	是	NULL		读者电话
email	varchar(100)	gb2312_chinese_ci	是	NULL		E-mail地址
createDate	date		是	NULL		办卡时间
operator	varchar(30)	gb2312_chinese_ci	是	NULL		操作员
remark	mediumtext	gb2312_chinese_ci	是	NULL		备注
typeid	int(11)		是	NULL		读者类型

图 6.15　读者信息表结构



限于篇幅，笔者在此只给出较重要的数据表，其他数据表参见本书附带的光盘。





6.5　首 页 设 计

6.5.1　首页概述

管理员通过系统登录模块的验证后，可以登录到图书馆管理系统的首页。系统首页主要包括导航栏、排行榜和版权信息 3 部分。其中，导航栏中的功能菜单将根据登录管理员的权限进行显示。例如，系统管理员 Tsoft 登录后，将拥有整个系统的全部功能，因为它是超级管理员。

本案例中提供的系统首页如图 6.16 所示。该页面在本书光盘中的路径为\TM\06\library\index.php。



图 6.16　学校图书馆管理系统首页

6.5.2　首页技术分析

学校图书馆管理系统是一个功能全面、大型的 Web 网站，通过对网站的安全性考虑，本网站对该系统进行权限的分配，只有管理员级别的超级用户可以对普通用户的权限进行管理和设置。系统首页主要通过判断管理员的权限来显示该用户所操作的功能模块，关键代码如下：

例程01　代码位置：光盘\TM\06\library\navigation.php

```
<?php
session_start();                                          //初始化 session 变量
include("conn/conn.php");                                 //连接数据库文件
$query=mysql_query("select m.id,m.name,p.id,p.sysset,p.readerset,p.bookset,p.borrowback,p.sysquery from tb_manager
as m left join (select * from tb_purview ) as p on m.id=p.id where name='$_SESSION[admin_name]'");
$info=mysql_fetch_array($query);                          //检索用户权限
?>
<!--检索用户所对应的权限，如果权限值为 1，则说明该功能可用，并输出到浏览器，否则不显示-->
<td width="70%" align="right">
<a href="index.php" class="a1">首页</a> |
 <?php if($info[sysset]==1){ ?><a onmouseover=showmenu(event,sysmenu) onmouseout=delayhidemenu()
style="CURSOR:hand"  class="a1">系统设置</a> | <?php } ?>
<?php if($info[readerset]==1){?><a onmouseover=showmenu(event,readermenu) onmouseout=delayhidemenu()
style="CURSOR:hand" class="a1">读者管理</a> | <?php } ?>
<?php if($info[bookset]==1){ ?><a href="book.php" class="a1">图书档案管理</a> | <?php }?>
```

```
<?php if($info[borrowback]==1){?><a  onmouseover=showmenu(event,borrowmenu) onmouseout=delayhidemenu()
style="CURSOR:hand"class="a1" >图书借还</a>  ┆  <?php }?>
<?php if($info[sysquery]==1){ ?><a onmouseover=showmenu(event,querymenu) onmouseout=delayhidemenu()
style="CURSOR:hand" class="a1">系统查询</a>  ┆  <?php } ?>
<a   href="pwd_Modify.php" class="a1">更改口令</a>  ┆
<a href="safequit.php" class="a1">注销</a>
</td>
```

说明 在权限信息表 tb_purview 中，权限值为 1，代表具备该模块的操作权限；权限值为 0，代表不具备该模块的操作权限。

在实现系统导航菜单时，引用了 JavaScript 文件 menu.JS，该文件中包含全部实现半透明背景菜单的 JavaScript 代码。

技巧 将页面中所涉及的 JavaScript 代码保存在一个单独的 JS 文件中，然后通过<script></script>将其引用到需要的页面，可以规范页面代码。在系统导航页面引用 menu.JS 文件的代码如下：

```
<script src="JS/menu.JS"></script>
```

6.5.3 首页的实现过程

系统首页的内容显示区用于显示图书的排行信息，并将排行结果按借阅数量降序排列。该页的关键代码如下：

例程02　代码位置：光盘\TM\06\library\index.php

```
<?php
include("conn/conn.php");                                             //连接数据源文件
$sql=mysql_query("select * from (select bookid,count(bookid) as degree from tb_borrow group by bookid) as
borr join (select b.*,c.name as bookcasename,p.pubname,t.typename from tb_bookinfo b left join tb_bookcase
c on b.bookcase=c.id join tb_publishing p on b.ISBN=p.ISBN join tb_booktype t on b.typeid=t.id where b.del=0)
as book on borr.bookid=book.id order by borr.degree desc limit 10");
$info=mysql_fetch_array($sql);                                        //检索图书借阅信息
$i=1;
do{                                                                   //应用 do…while 循环语句显示图书信息
?>
<tr>
     <td height="25" align="center"><?php echo $i;?></td>
     <td style="padding:5px;"> <?php echo $info[barcode];?></td>
     <td style="padding:5px;"><?php echo $info[bookname];?></td>
     <td style="padding:5px;"><?php echo $info[typename];?></td>
     <td align="center"> <?php echo $info[bookcasename];?></td>
     <td align="center"> <?php echo $info[pubname];?></td>
     <td align="center"><?php echo $info[author];?></td>
     <td align="center"><?php echo $info[price];?></td>
     <td align="center"><?php echo $info[degree];?></td>
```



```
</tr>
<?php
 $i=$i+1;                                            //变量自加 1 操作
 }while($info=mysql_fetch_array($sql));              //do…while 循环语句结束
?>
```

6.6　管理员模块设计

6.6.1　管理员模块概述

管理员模块主要包括管理员登录、查看管理员列表、添加管理员信息、管理员权限设置、管理员删除和更改口令等 6 个功能。管理员模块的框架如图 6.17 所示。

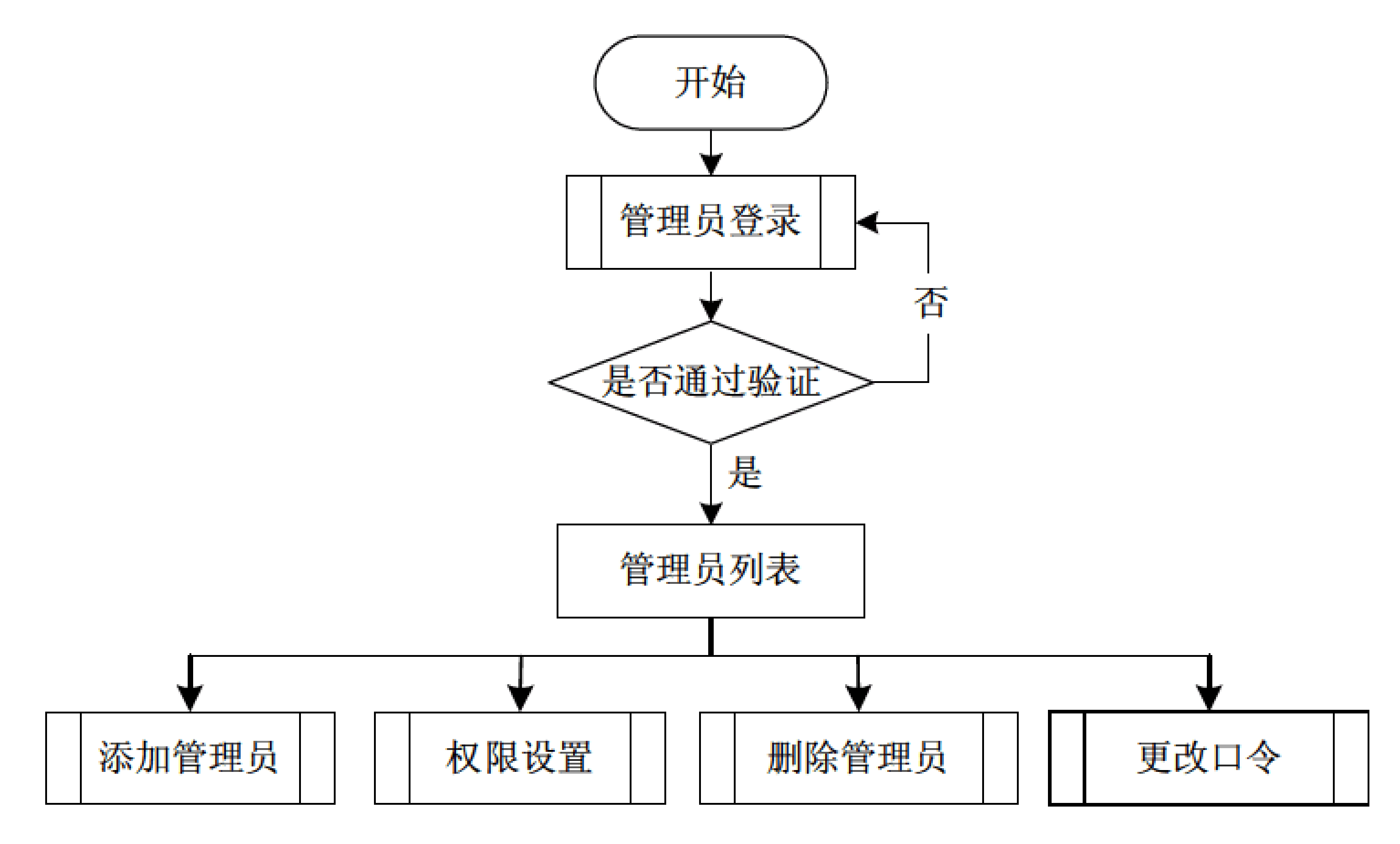


图 6.17　管理员模块的框架图

6.6.2　管理员模块技术分析

在管理员模块中，涉及的数据表是 tb_manager（管理员信息表）和 tb_purview（权限表）。其中，管理员信息表中保存的是管理员名称和密码等信息，权限表中保存的是各管理员的权限信息，这两个表通过各自的 id 字段相关联。通过这两个表可以获得完整的管理员信息。

注意　在实现系统登录前，需要在 MySQL 数据库中手动添加一条系统管理员的数据（管理员名为 Tsoft，密码为 111，拥有所有权限），即在 MySQL 的客户端命令行中应用下面的语句分别向管理员信息表 tb_manager 和权限表 tb_purview 中各添加一条数据。

```
#添加管理员信息
insert into tb_manager (name,pwd) values('Tsoft','111');
#添加权限信息
insert into tb_purview values(1,1,1,1,1,1);
```

从网站安全的角度考虑，仅仅有上面介绍的系统登录页面并不能有效地保证系统的安全，一旦系统首页面的地址被他人获得，就可以通过在地址栏中输入系统的首页面地址而直接进入到系统中。为了便于网站的维护，因此将验证用户是否登录的代码封装在独立的 PHP 文件中，即 check_login.php 文件。验证用户是否登录的具体代码如下：

例程03　代码位置：光盘\TM\06\library\check_login.php

```
<?php
session_start();                                         //初始化 session 变量
if($_SESSION[admin_name]==""){                           //如果 session 变量为空，则说明用户未登录
    echo "<script>alert('对不起，请通过正确的途径登录博考图书馆管理系统!');window.location.href='login.php';</script>";
}
?>
```

当系统调用首页时，会判断 session 变量 admin_name 是否存在，如果不存在，将页面重定向到系统登录（login.php）页面。

6.6.3　系统登录的实现过程

系统登录使用的数据表：tb_manager

系统登录是进入学校图书馆管理系统的入口，主要用于验证管理员的身份。运行本系统，首先进入的是系统登录页面，在该页面中，系统管理员可以通过输入正确的管理员名称和密码登录到系统首页，当用户没有输入管理员名称或密码时，系统会通过 JavaScript 进行判断，并给予信息提示。系统登录页面的运行结果如图 6.18 所示。



图 6.18　系统登录页面的运行结果

系统登录页面主要用于收集管理员的输入信息及通过自定义的 JavaScript 函数验证输入信息是否为空。该页面中所涉及的表单元素如表 6.1 所示。

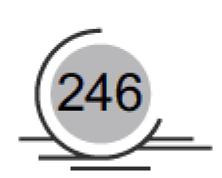
表 6.1　系统登录页面所涉及的表单元素

名　称	元素类型	重要属性	含　义
form1	form	method="post" action="chklogin.php"	管理员登录表单
name	text	size="25"	管理员名称
pwd	password	size="25"	管理员密码
submit	submit	value="确定" onclick="return check(form1)"	“确定”按钮
submit3	reset	value="重置"	“重置”按钮
submit2	button	value="关闭" onClick="window.close();"	“关闭”按钮

编写自定义的 JavaScript 函数，用于判断管理员名称和密码是否为空。代码如下：

例程04　代码位置：光盘\TM\06\library\login.php

```
<script language="javascript">
function check(form){                        //自定义一个 JavaScript 函数 check()
```



```
        if (form.name.value==""){                      //如果管理员名称为空，则弹出提示信息，并重新返回焦点
            alert("请输入管理员名称!");form.name.focus();return false;
        }
        if (form.pwd.value==""){                       //如果管理员密码为空，则弹出提示信息，并重新返回焦点
            alert("请输入密码!");form.pwd.focus();return false;
        }
}
</script>
```

提交表单到数据处理页，页面中为了防止非法用户进入学校图书馆管理系统首页，通过调用类的chkinput()方法实现判断用户名和密码是否正确。如果为合法用户，则可以登录学校图书馆管理系统的首页；否则，弹出相应的错误提示。关键代码如下：

例程05　代码位置：光盘\TM\06\library\chklogin.php

```
<?php
session_start();                                    //初始化 session 变量
$A_name=$_POST[name];                               //接收表单提交的用户名
$A_pwd=$_POST[pwd];                                 //接收表单提交的密码
❶   class chkinput{                                 //定义类
    var $name;
    var $pwd;
    function chkinput($x,$y){                       //定义一个方法
      $this->name=$x;                               //将管理员名称传给类对象$this->name
      $this->pwd=$y;                                //将管理员密码传给类对象$this->pwd
     }
    function checkinput(){
      include("conn/conn.php");                     //连接数据库文件
      $sql=mysql_query("select * from tb_manager where name='".$this->name."' and pwd='".$this->pwd."'",$conn);
      $info=mysql_fetch_array($sql);                //检索管理员名称和密码是否正确
      if($info==false){                             //如果管理员名称或密码不正确，则弹出相关提示信息
           echo "<script language='javascript'>alert('您输入的管理员名称错误，请重新输入！'); history.back();
</script>";
           exit;
       }
      else{                                         //如果管理员名称或密码正确，则弹出相关提示信息
         echo "<script>alert('管理员登录成功!');window.location='index.php';</script>";
         $_SESSION[admin_name]=$info[name];  //将管理员名称存到$_SESSION[admin_name]变量中
         $_SESSION[pwd]=$info[pwd];          //将管理员密码存到$_SESSION[pwd]变量中
    }
  }
}
❷           $obj=new chkinput(trim($name),trim($pwd));  //创建对象
❸      $obj->checkinput();                           //调用类
?>
```

代码贴士

❶ class：创建一个 PHP 类时，必须使用关键字 class 进行声明，该关键字后紧跟类的名称，之后用大括号将类体进行封装。

❷ new：在 PHP 中应用 new 关键字创建对象。该模块创建了一个名为 chkinput 的验证管理员类。

❸ $obj->checkinput();：通过$obj 对象调用 checkinput 类中的属性和方法，即管理员名称和密码。

6.6.4 查看管理员的实现过程

查看管理员使用的数据表：tb_manager、tb_purview

管理员登录后，选择“系统设置”/“管理员设置”菜单项，进入到查看管理员列表页面。在该页面中，将以表格的形式显示全部管理员及其权限信息，并提供添加管理员信息、删除管理员信息和设置管理员权限的超链接。查看管理员列表页面的运行结果如图 6.19 所示。



图 6.19 查看管理员列表页面的运行结果

首先使用左外联接语句（left join…on）从数据表 tb_manager 和 tb_purview 中查询出符合条件的数据，然后将查询结果应用 do…while 循环语句输出到浏览器。关键代码如下：

例程06 代码位置：光盘\TM\02\library\manager.php

```
<?php
include("conn/conn.php");                                    //连接数据库文件
❶   $sql=mysql_query("select m.id,m.name,p.sysset,p.readerset,p.bookset,p.borrowback,p.sysquery from
tb_manager as m left join (select * from tb_purview) as p on m.id=p.id");
$info=mysql_fetch_array($sql);                               //检索数据信息
do{                                                          //应用 do…while 循环语句输出查询结果
?>
/* ********************************输出符合查询条件的记录********************************* */
<tr>
    <td style="padding:5px;"><?php echo $info[name];?></td>
❷   <td align="center"><input name="checkbox" type="checkbox" class="noborder" value="checkbox"
disabled="disabled" <?php if($info[sysset]==1){echo ("checked");}?>></td>
    <td align="center"><input name="checkbox" type="checkbox" class="noborder" value="checkbox"
 disabled="disabled" <?php if($info[readerset]==1){echo("checked");}?>></td>
    <td align="center"><input name="checkbox" type="checkbox" class="noborder" value="checkbox" disabled
<?php if($info[bookset]==1){echo("checked");}?>></td>
    <td align="center"><input name="checkbox" type="checkbox" class="noborder" value="checkbox" disabled
<?php if($info[borrowback]==1){echo("checked");}?>></td>
    <td align="center"><input name="checkbox" type="checkbox" class="noborder" value="checkbox" disabled
<?php if($info[sysquery]==1){echo("checked");}?>></td>
    <td align="center"><a href="#" onClick="window.open('manager_modify.php?id=<?php echo
$info[id]; ?>','','width=292,height=175')">权限设置</a></td>
    <td align="center"><a href="manager_del.php?id=<?php echo $info[id];?>">删除</a></td>
  </tr>
/* ************************************************************************************* */
```



```
<?php
  }while($info=mysql_fetch_array($sql));                    //do…while 循环语句结束
?>
```

代码贴士

❶ left join…on：应用左外联接将两个表或多个表连接起来，返回部分或全部匹配行，详解参见 6.10.2 节。

❷ <?php if($info[sysset]==1){echo ("checked");}?>：如果系统设置字段的值为 1，则复选框处于选中状态。

6.6.5　添加管理员的实现过程

添加管理员使用的数据表：tb_manager

管理员登录后，选择“系统设置”/“管理员设置”菜单项，进入到查看管理员列表页面，在该页面中单击“添加管理员信息”超链接，打开添加管理员信息页面。添加管理员信息页面的运行结果如图 6.20 所示。

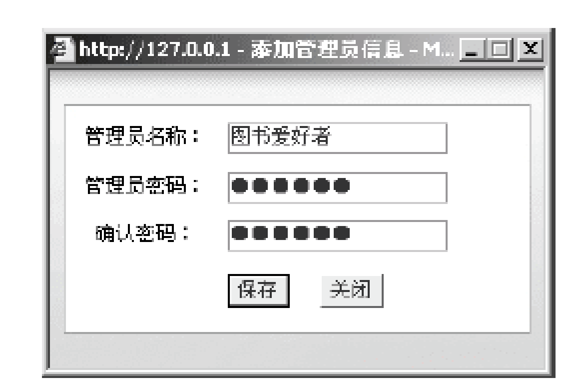


图 6.20　添加管理员页面的运行结果



注意　新添加的管理员信息没有权限，必须通过设置管理员权限为其指定可操作的功能模块。

在查看管理员列表页面，单击“添加管理员信息”超链接文字的 HTML 代码如下：

例程07　代码位置：光盘\TM\06\library\manager.php

```
<a href="#" onClick="window.open('manager_add.php','','width=292,height=175')">添加管理员信息</a>
```

添加管理员页面主要用于收集输入的管理员信息及通过自定义的 JavaScript 函数验证输入信息是否合法。该页面中所涉及的表单元素如表 6.2 所示。

表 6.2　添加管理员页面所涉及的表单元素

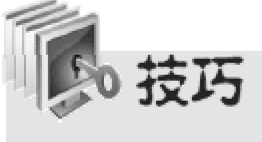
名　称	元 素 类 型	重 要 属 性	含　义
form1	form	method="post" action="manager_ok.php"	表单
name	text		管理员名称
pwd	password		管理员密码
pwd1	password		确认密码
submit	submit	value="保存" onClick="check(form1)"	“保存”按钮
Submit2	button	value="关闭" onClick="window.close();"	“关闭”按钮

在添加管理员页面中，输入合法的管理员名称及密码后，单击“保存”按钮，提交表单信息到数据处理页，将添加的管理员信息保存到数据表中。如果添加成功，弹出成功的提示信息；否则，弹出错误提示。代码如下：

例程08　代码位置：光盘\TM\02\library\manager_ok.php

```
<?php
include("conn/conn.php");                         //连接数据库文件
if($_POST[submit]!=""){                           //如果单击了“保存”按钮，则执行下面的操作
$name=$_POST[name];                               //获取管理员名称
```

```
$pwd=$_POST[pwd];                                    //获取管理员密码
$sql=mysql_query("insert into tb_manager (name,pwd) values('$name','$pwd')");
if($sql==true){                                      //向数据表中添加管理员信息成功，则给出提示信息
echo "<script language=javascript>alert('管理员添加成功！');window.close();window.opener.location.reload();
</script>";
}
else{                                                //向数据表中添加管理员信息失败，则给出提示信息
echo "<script language=javascript>alert('管理员添加失败！');window.close();window.opener.location.reload();
</script>";
}
}
?>
```



在添加管理员处理页中应用“window.opener.location.reload();”语句刷新父窗口中的信息。

6.6.6　设置管理员权限的实现过程

设置管理员权限使用的数据表：tb_manager、tb_purview

在查看管理员列表页面单击指定管理员后面的“权限设置”超链接，即可进入到“权限设置”页面，设置该管理员的操作权限。权限设置页面的运行结果如图 6.21 所示。

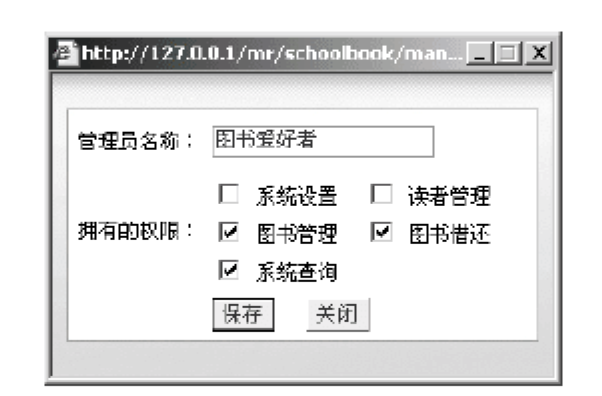


图 6.21　权限设置页面的运行结果

权限设置页面中所涉及的表单元素如表 6.3 所示。

表 6.3　权限设置页面所涉及的表单元素

名　称	元素类型	重要属性	含　义
form1	form	method="post" action="manager_modifyok.php"	表单
id	hidden	value="<?php echo $info[id];?>"	管理员编号
name	text	value="<?php echo $info[name];?>"	管理员名称
sysset	checkbox	<?php if($info[sysset]==1){ echo("checked");}?>	系统设置
readerset	checkbox	<?php if($info[readerset]==1){ echo("checked");}?>	读者管理
bookset	checkbox	<?php if($info[bookset]==1){echo("checked");}?>	图书管理
borrowback	checkbox	<?php if($info[borrowback]==1){echo("checked");}?>	图书借还
sysquery	checkbox	<?php if($info[sysquery]==1){echo("checked");}?>	系统查询
submit	submit	class="btn_grey"	“保存”按钮
Submit2	button	value="关闭" onClick="window.close();"	“关闭”按钮

在查看管理员列表页面中，添加权限设置列，并在该列中添加以下用于打开权限设置页面的超链接代码。

例程09　代码位置：光盘\TM\02\library\manager.php

```
<a href="#" onClick="window.open('manager_modify.php?id=<?php echo $info[id]; ?>','','width=292,height=
175')">权限设置</a>
```





从上面的 URL 地址中可以获取设置管理员权限页所涉及的 id 号，将 id 号提交给处理页 manager_modifyok.php，修改 id 号所对应的管理员信息。具体代码如下：

例程10　代码位置：光盘\TM\02\library\manager_modifyok.php

```
<?php
include("conn/conn.php");                                   //连接数据库文件
if($_POST[submit]!=""){                                     //如果提交表单，则执行以下操作
$id=$_POST[id];                                             //获取 id 信息
❶  $sysset=$_POST[sysset]==""?0:1;                          //应用三目运算符求出“系统设置”复选框的值
$readerset=$_POST[readerset]==""?0:1;                       //应用三目运算符求出“读者管理”复选框的值
$bookset=$_POST[bookset]==""?0:1;                           //应用三目运算符求出“图书管理”复选框的值
$borrowback=$_POST[borrowback]==""?0:1;                     //应用三目运算符求出“图书借还”复选框的值
$sysquery=$_POST[sysquery]==""?0:1;                         //应用三目运算符求出“系统查询”复选框的值
$query=mysql_query("select * from tb_purview where id=$id");
    $info=mysql_fetch_array($query);                        //检索权限信息表中是否存在该管理员
if($info==false){                                           //如果不存在，向权限表中添加管理员权限信息
❷  mysql_query("insert into tb_purview(id,sysset,readerset,bookset,borrowback,sysquery)
values($id,$sysset,$readerset,$bookset,$borrowback,$sysquery)");
}
else{                                                       //否则，更新管理员的权限信息
    mysql_query("update tb_purview set
 sysset=$sysset,readerset=$readerset,bookset=$bookset,borrowback=$borrowback,sysquery=$sysquery
where id='$id'");
}
echo"<script language=javascript>alert('权限设置修改成功！');window.close();window.opener.location.reload();
</script>";                                                 //更新成功，弹出提示信息，并更新父窗口
}
?>
```

代码贴士

❶ $_POST[sysset]==""?0:1;：应用三目运算符求出“系统设置”复选框的值，如果等于空，值为 0，否则值为 1。
❷ insert…into：向权限信息表中添加一行数据信息，insert…into 语句只适用于对单行数据的插入。

6.6.7　删除管理员的实现过程

删除管理员使用的数据表：tb_manager、tb_purview

在查看管理员列表页面中，单击指定管理员信息后面的“删除”超链接，该管理员及其权限信息将被删除。

在查看管理员列表页面中添加以下用于删除管理员信息的超链接代码。

例程11　代码位置：光盘\TM\06\library\manager.php

```
<a href="manager_del.php?id=<?php echo $info[id];?>">删除</a>
```

从上面的 URL 地址中，可以获取删除管理员所涉及的 id 号，将 id 号提交给 manager_del.php 处理页删除 id 号所对应的管理员信息。具体代码如下：

例程12　代码位置：光盘\TM\06\library\manager_del.php

```
<?php
include("conn/conn.php");                                   //连接数据库文件
```

```
$id=$_GET[id];                                             //获取管理员的 id 号
$sql=mysql_query("delete from tb_manager where id='$id'");  //删除管理员表中 id 号所对应的管理员信息
$query=mysql_query("delete from tb_purview where id='$id'"); //删除权限表中 id 号所对应的管理员权限
if($sql==true and $query==true ){                          //如果删除操作成功，则弹出提示信息
    echo "<script language=javascript>alert('管理员删除成功！');history.back();</script>";
}
else{                                                      //如果删除操作失败，则弹出提示信息
    echo "<script language=javascript>alert('管理员删除失败！');history.back();</script>";
}
?>
```

6.6.8 单元测试

在开发完管理员模块后，需要对该模块进行单元测试。当管理员修改操作员的权限时，该操作员的权限没有改变。下面给出修改权限设置处理页的源代码：

```
<?php
include("conn/conn.php");                                  //连接数据库文件
if($_POST[submit]!=""){                                    //如果提交表单，则执行以下操作
$id=$_POST[id];                                            //获取 id 信息
$sysset=$_POST[sysset];                                    //获取“系统设置”复选框的值
$readerset=$_POST[readerset];                              //获取“读者管理”复选框的值
$bookset=$_POST[bookset];                                  //获取“图书管理”复选框的值
$borrowback=$_POST[borrowback];                            //获取“图书借还”复选框的值
$sysquery=$_POST[sysquery];                                //获取“系统查询”复选框的值
$query=mysql_query("select * from tb_purview where id=$id");
$info=mysql_fetch_array($query);                           //检索权限信息表中是否存在该管理员
if($info==false){                                          //如果不存在，向权限表中添加管理员权限信息
    mysql_query("insert into tb_purview(id,sysset,readerset,bookset,borrowback,sysquery)
values($id,$sysset,$readerset,$bookset,$borrowback,$sysquery)");
}
else{                                                      //否则，更新管理员的权限信息
    mysql_query("update tb_purview set
sysset=$sysset,readerset=$readerset,bookset=$bookset,borrowback=$borrowback,sysquery=$sysquery
where id='$id'");
}
echo"<script language=javascript>alert('权限设置修改成功！');window.close();window.opener.location.reload();
</script>";                                                //更新成功，弹出提示信息，并更新父窗口
}
?>
```

为了找出错误的原因，笔者应用 echo()语句对$sysset 等一系列传递的值进行输出，输出的结果显示$sysset 的值始终等于 0，这充分说明在修改复选框时，没有获取到选中复选框的值。

解决该问题的方法是对提交的复选框的状态进行判断，如果复选框处于未选中状态，则值为 0；如果处于选中状态，则值为 1。这样就可以获取到复选框选中状态的值了。

处理复选框的状态有两种方法：一种是应用 if 条件语句对逐个值进行判断，判断复选框的当前状态，并赋予其值（但这种方法过于繁琐）；另一种是应用三目运算符，简单快捷地计算复选框的值（这



种方法可以简化代码，本模块采用该方法获取复选框的值）。

应用三目运算符获取复选框的值的代码如下：

```
$sysset=$_POST[sysset]==""?0:1;                 //应用三目运算符求出“系统设置”复选框的值
$readerset=$_POST[readerset]==""?0:1;           //应用三目运算符求出“读者管理”复选框的值
$bookset=$_POST[bookset]==""?0:1;               //应用三目运算符求出“图书管理”复选框的值
$borrowback=$_POST[borrowback]==""?0:1;         //应用三目运算符求出“图书借还”复选框的值
$sysquery=$_POST[sysquery]==""?0:1;             //应用三目运算符求出“系统查询”复选框的值
```

完整的权限设置处理页的代码参见例程 10。

6.7　图书档案管理模块设计

6.7.1　图书档案管理模块概述

图书档案管理模块主要包括查看图书列表、添加图书信息、修改图书信息、删除图书信息和查看图书详细信息等 5 个功能。图书档案模块的框架如图 6.22 所示。

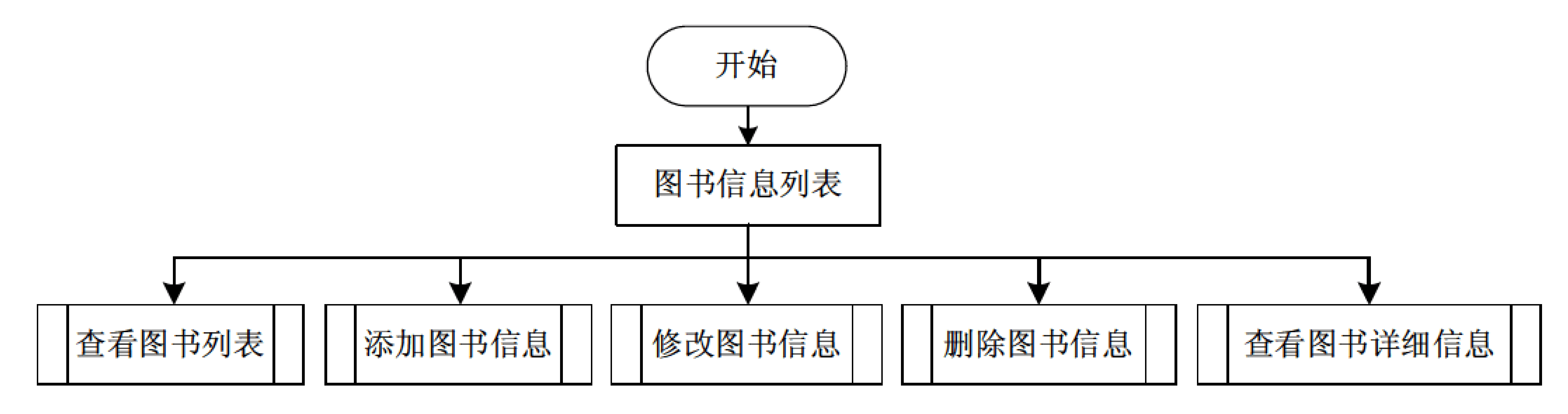


图 6.22　图书档案模块的框架图

6.7.2　图书档案管理模块技术分析

在图书档案管理模块中，涉及的数据表是 tb_bookinfo（图书信息表）、tb_bookcase（书架设置表）、tb_booktype（图书类型表）和 tb_publishing（出版社信息表），这 4 个数据表间通过相应的字段进行关联，如图 6.23 所示。通过以上 4 个表可以获得完整的图书档案信息。

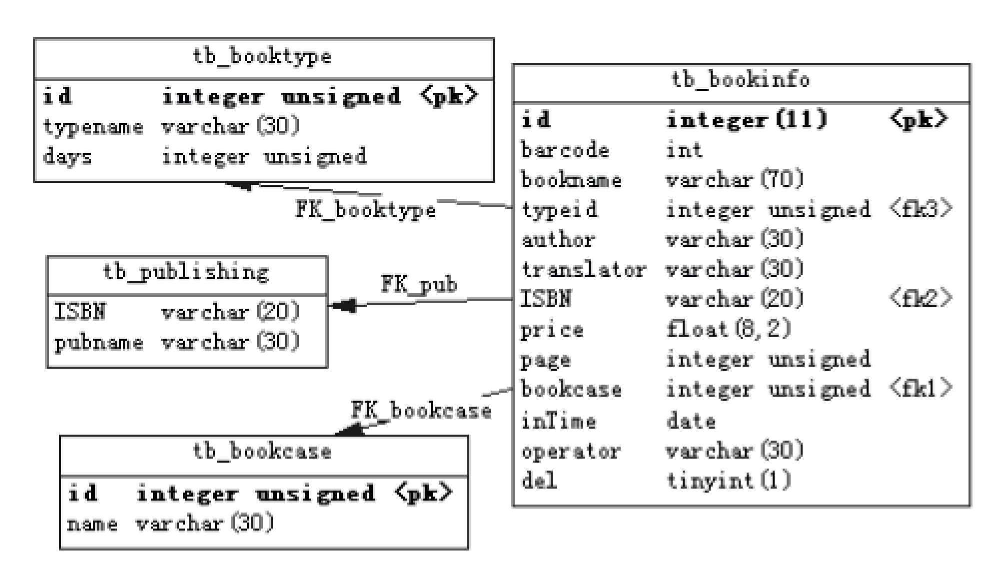


图 6.23　图书档案管理模块各表间关系图



注意 本模块主要应用联接语句实现多表查询，关于联接语句的详细讲解参见 6.10 节。

6.7.3 查看图书信息列表的实现过程

查看图书信息列表使用的数据表：tb_bookinfo、tb_bookcase、tb_booktype、tb_publishing

管理员登录后，选择“图书管理”/“图书档案管理”菜单项，进入到查看图书列表页面，在该页面中将显示全部图书信息列表，同时提供添加图书信息、删除图书信息、修改图书信息的超链接。查看图书信息列表页面的运行结果如图 6.24 所示。



图 6.24 查看图书信息列表的运行结果

打开功能导航 navigation.php 文件，设置“图书档案管理”菜单项的超链接的代码如下：

例程13 代码位置：光盘\TM\02\library\navigation.php

```
<a href="book.php" class="a1">图书档案管理</a>
```

首先应用 join…on 内联接语句将 tb_bookinfo、tb_bookcase、tb_booktype 和 tb_publishing 4 个数据表连接起来检索指定条件的图书信息，然后应用 do…while 循环语句输出查询结果到浏览器。查看图书信息页面的代码如下：

例程14 代码位置：光盘\TM\06\library\book.php

```
<?php
include("conn/conn.php");                          //连接数据库文件
$query=mysql_query("select book.barcode,book.id as bookid,book.bookname,bt.typename,pb.pubname, bc.
name from tb_bookinfo book join tb_booktype bt on book.typeid=bt.id join tb_publishing pb on book.ISBN=
pb.ISBN join tb_bookcase bc on book.bookcase=bc.id");
$result=mysql_fetch_array($query);                 //应用外联接检索图书信息
?>
…                                                  //省略图书信息标题 HTML 标记部分
<?php
 do{                                               //应用 do…while 循环语句输出查询结果
?>
   <tr>
     <td style="padding:5px;"> <?php echo $result[barcode];?></td>
     <td style="padding:5px;"><a href="book_look.php?id=<?php echo $result[bookid];?>"><?php echo
$result[bookname];?></a></td>
```



```
      <td style="padding:5px;"> <?php echo $result[typename];?></td>
      <td style="padding:5px;"> <?php echo $result[pubname];?></td>
      <td style="padding:5px;"> <?php echo $result[name];?></td>
      <td align="center"><a href="book_Modify.php?id=<?php echo $result[bookid];?>">修改</a></td>
      <td align="center"><a href="book_del.php?id=<?php echo $result[bookid];?>">删除</a></td>
   </tr>
<?
   }while($result=mysql_fetch_array($query));            //do…while 循环语句结束
?>
```

注意　关于 join…on 内联接语句的使用方法参见 6.10.1 节。

6.7.4　添加图书信息的实现过程

添加图书信息使用的数据表：tb_bookinfo、tb_bookcase、tb_booktype、tb_publishing

管理员登录系统后，在导航栏中单击“图书档案管理”超链接，进入到查看图书列表页面。在该页面中单击“添加图书信息”超链接，进入到添加图书信息页面。添加图书信息页面的运行结果如图 6.25 所示。

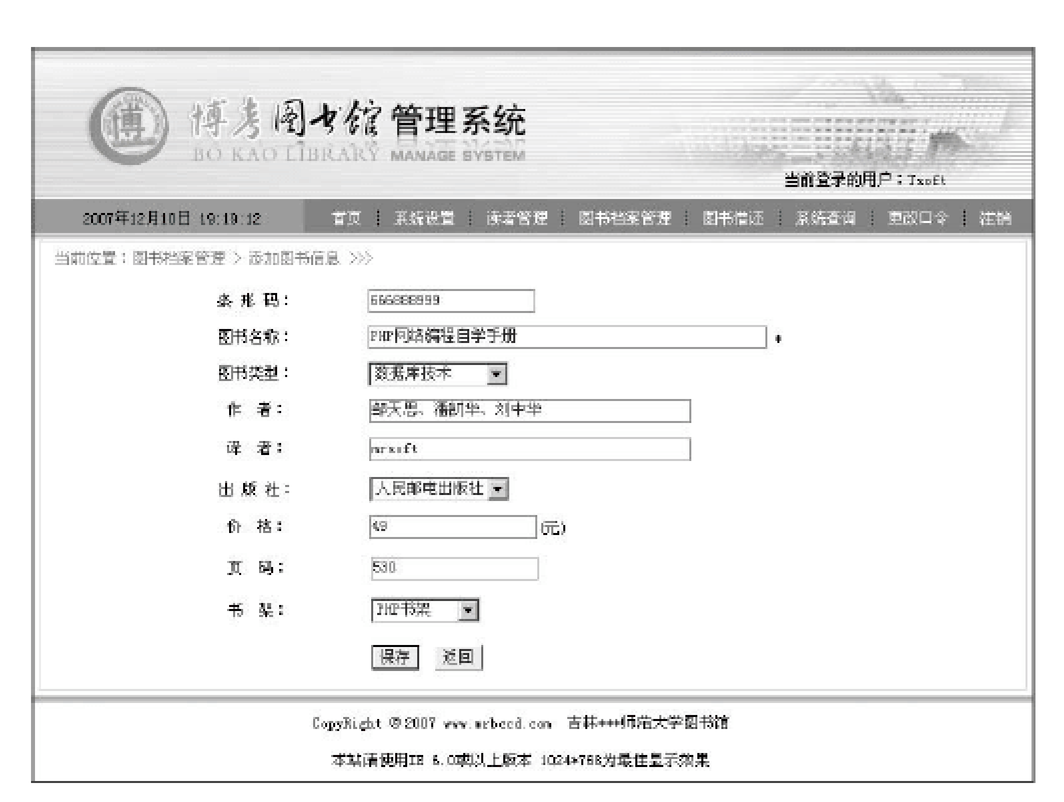


图 6.25　添加图书信息页面的运行结果

在查看图书列表页面中，设置“添加图书信息”超链接的代码如下：

例程15　代码位置：光盘\TM\06\library\book.php

```
<a href="book_add.php">添加图书信息</a>
```

添加图书信息页面主要用于收集输入的图书信息以及通过自定义的 JavaScript 函数验证输入信息是否合法。该页面中所涉及的重要表单元素如表 6.4 所示。

表 6.4　添加图书信息页面所涉及的重要表单元素

名　　称	元 素 类 型	重 要 属 性	含　　义
form1	form	method="post" action="book_ok.php"	表单
typeId	select	<?php include("Conn/conn.php"); $sql=mysql_query("select * from tb_booktype");	图书类型

续表

名　称	元素类型	重要属性	含　义
typeId	select	$info=mysql_fetch_array($sql); do{ ?> <option value="<?php echo $info[id];?>"> <?php echo $info[typename];?></option> <?php }while($info=mysql_fetch_array($sql));?>	图书类型
isbn	select	<?php $sql2=mysql_query("select * from tb_publishing"); $info2=mysql_fetch_array($sql2); do{ ?> <option value="<?php echo $info2[ISBN];?>"> <?php echo $info2[pubname];?></option> <?php }while($info2=mysql_fetch_array($sql2));?>	出版社
bookcaseid	select	<?php $sql3=mysql_query("select * from tb_bookcase"); $info3=mysql_fetch_array($sql3); do{ ?> <option value="<?php echo $info3[id];?>"> <?php echo $info3[name];?></option> <?php }while($info3=mysql_fetch_array($sql3));?>	书架名称
operator	hidden	value="<?php echo $info3[name];?>"	操作员
Submit	submit	onClick="return check(form1)"	“保存”按钮
Submit2	button	onClick="history.back();"	“返回”按钮

由于添加图书信息的方法同添加管理员信息的方法类似，所以此处只给出向图书信息表中插入数据的 SQL 语句，详细代码参见光盘。向图书信息表中插入数据的 SQL 语句如下：

例程16　代码位置：光盘\TM\06\library\book_ok.php

```
mysql_query("insert into
tb_bookinfo(barcode,bookName,typeid,author,translator,ISBN,price,page,bookcase,inTime,operator )values('$b
arcode','$bookName','$typeid','$author','$translator','$isbn','$price','$page','$bookcaseid','$inTime','$operator')");
```

6.7.5　修改图书信息的实现过程

修改图书信息使用的数据表：tb_bookinfo、tb_bookcase、tb_booktype、tb_publishing

管理员登录系统后，在导航栏中单击“图书档案管理”超链接，进入到查看图书列表页面。单击想要修改的图书信息后面的“修改”超链接，进入到“修改图书信息”页面。修改图书信息页面的运行结果如图 6.26 所示。





图 6.26　修改图书信息页面的运行结果

在图书信息列表页面中，添加“修改”超链接的代码如下：

例程17　代码位置：光盘\TM\06\library\book.php

```
<a href="book_Modify.php?id=<?php echo $result[bookid];?>">修改</a>
```

在修改图书信息页面中修改图书信息后，单击“保存”按钮，提交表单信息到数据处理页 book_Modify_ok.php，应用 UPDATE 语句将修改的图书信息保存到数据表 tb_bookinfo 中，并弹出“图书信息修改成功！”提示信息，将页面重定向到修改图书信息页。数据处理页的代码如下：

例程18　代码位置：光盘\TM\06\library\book_Modify_ok.php

```
<?php
session_start();                                          //初始化 session 变量
include("conn/conn.php");                                 //连接数据库文件
$bid=$_POST[bid];                                         //获取图书 id 号
$operator=$_SESSION[admin_name];                          //获取管理员名称
$barcode=$_POST[barcode];                                 //获取图书条形码
$bookName=$_POST[bookName];                               //获取图书名称
$typeid=$_POST[typeId];                                   //获取图书类型 id 号
$author=$_POST[author];                                   //获取图书作者
$translator=$_POST[translator];                           //获取图书译者
$isbn=$_POST[isbn];                                       //获取出版社 ISBN
$price=$_POST[price];                                     //获取图书单价
$page=$_POST[page];                                       //获取图书页码
$bookcase=$_POST[bookcaseid];                             //获取图书书架 id 号
$inTime=date("Y-m-d");                                    //设置图书更新日期为当前日期
$query=mysql_query("update tb_bookinfo set barcode='$barcode', bookName='$bookName' , typeid='$typeid',
author='$author', translator='$translator', ISBN='$isbn' , price='$price' , page='$page' , bookcase='$bookcaseid',
inTime='$inTime', operator='$operator' where id=$bid");  //更新数据表
echo "<script language='javascript'>alert('图书信息修改成功!');history.back();</script>";
?>
```

6.7.6　删除图书信息的实现过程

删除图书信息使用的数据表：tb_bookinfo

在查看图书列表页面中，设置“删除”超链接的代码如下：

例程19　代码位置：光盘\TM\06\library\book.php

```
<a href="book_del.php?id=<?php echo $result[bookid];?>">删除</a>
```

单击想要删除的图书信息后面的“删除”超链接，提交表单信息到数据处理页 book_del.php，应用 DELETE 语句将指定的图书信息从数据表 tb_bookinfo 中删除，如果删除操作执行成功，则弹出“图书信息删除成功！”提示信息，并将页面重定向到图书信息列表页面。数据处理页的代码如下：

例程20　代码位置：光盘\TM\06\library\book_del.php

```
<?php
include("conn/conn.php");                                              //连接数据库文件
$info_del=mysql_query("delete from tb_bookinfo where id=$_GET[id]");   //删除指定的图书信息
if($info_del){                                                         //如果信息删除成功，则弹出提示
    echo "<script language='javascript'>alert('图书信息删除成功!');history.back();</script> ";
}
?>
```

6.8　图书借还模块设计

6.8.1　图书借还模块概述

图书借还模块主要包括图书借阅、图书续借、图书归还、图书档案查询、图书借阅查询、借阅到期提醒等 6 个功能。在图书借阅模块中的用户只有一种身份，那就是操作员，通过该身份可以进行图书借还等相关操作。图书借还模块的用例图如图 6.27 所示。

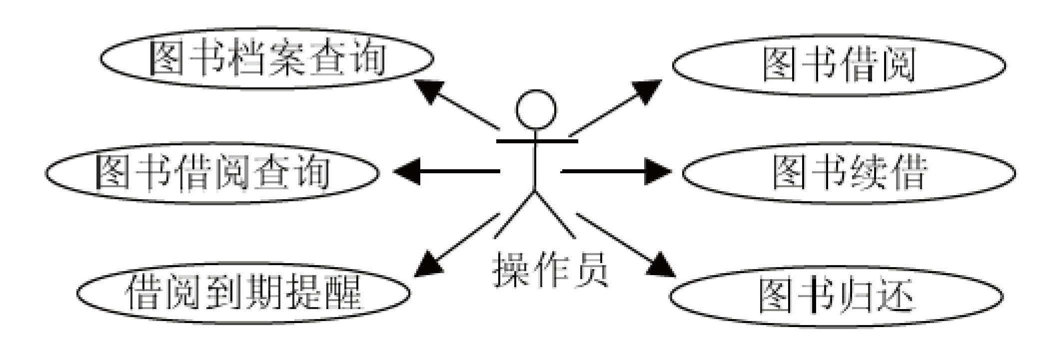


图 6.27　图书借还模块的用例图

6.8.2　图书借还模块技术分析

在图书借还模块中涉及的数据表是 tb_borrow（图书借阅信息表）、tb_bookinfo（图书信息表）和 tb_reader（读者信息表），这 3 个数据表间通过相应的字段进行关联，如图 6.28 所示。

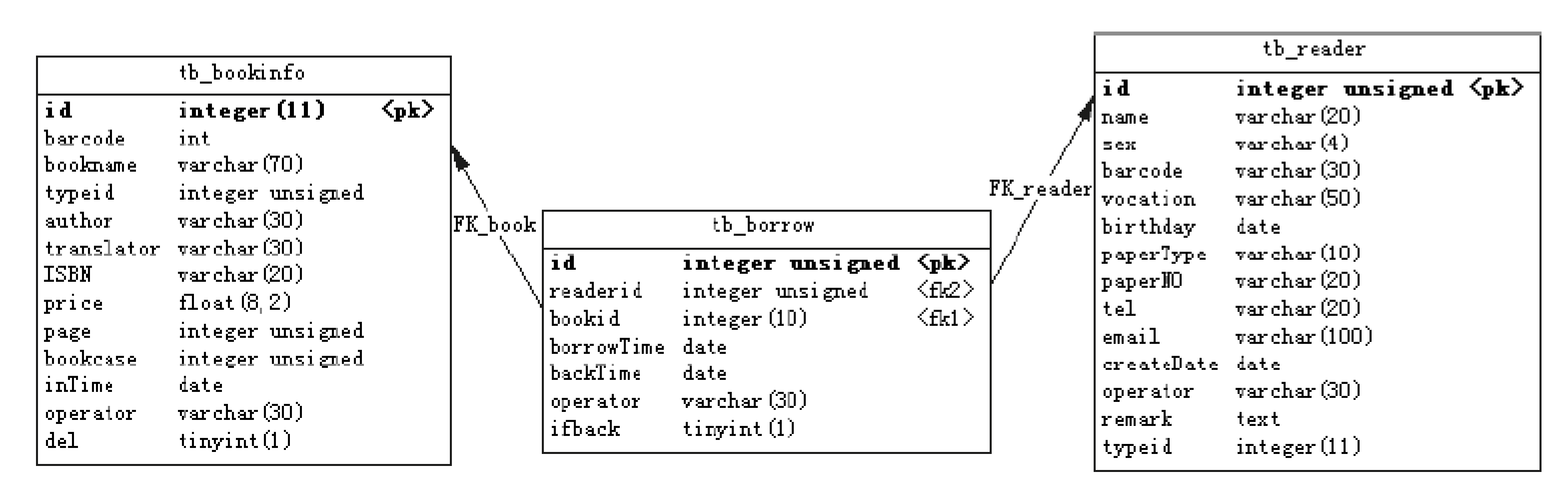


图 6.28　图书借还管理模块各表间关系图





注意　本模块主要应用联接语句实现多表查询，关于联接语句的详细讲解参见 6.10 节。

6.8.3　图书借阅的实现过程

图书借阅使用的数据表：tb_borrow、tb_bookinfo、tb_reader

管理员登录后，选择“图书借还”/“图书借阅”菜单项，进入到图书借阅页面，在该页面的“读者条形码”文本框中输入读者的条形码（如 123456789）后，单击“确定”按钮，系统会自动检索出该读者的基本信息和未归还的借阅图书信息。如果检索到对应的读者信息，将其显示在页面中，此时输入图书的条形码或图书名称后，单击“确定”按钮，借阅指定的图书，运行结果如图 6.29 所示。



图 6.29　图书借阅页面的运行结果

说明　当读者借阅图书完毕后，操作员通过单击“完成借阅”按钮，将重新载入图书借阅页面，当前页处于空信息状态，从而方便操作员对下一个读者进行借阅图书操作。

图书借阅页面总体上可以分为两个部分：一部分用于查询并显示读者信息；另一部分用于显示读者的借阅信息和添加读者借阅信息。图书借阅页面在 Dreamweaver 中的设计效果如图 6.30 所示。



图 6.30　图书借阅页面的设计效果

在进行图书借阅时，系统要求每个读者只能同时借阅一定数量的图书，并且该数量由读者类型表 tb_readertype 中的可借数量 number 决定，所以笔者编写了自定义的 checkbook() 函数，用于判断当前选择的读者是否还可以借阅新的图书，同时该函数还具有判断输入读者条形码或图书名称文本框是否为空的功能。代码如下：

例程21　代码位置：光盘\TM\02\bookBorrow.php

```
<script language="javascript">
function checkbook(form){                                    //自定义一个 JavaScript 函数 checkbook()
```

```
        if(form.barcode.value==""){                                   //如果读者条形码为空
                alert("请输入读者条形码!");form.barcode.focus();return; //弹出提示，焦点返回到条形码文本框
        }
        if(form.inputkey.value==""){                                  //如果图书查询文本框的值为空
                alert("请输入查询关键字!");form.inputkey.focus();return; //弹出提示，焦点返回到图书查询文本框
        }
        if(form.number.value-form.borrowNumber.value<=0){            //如果图书的借阅数量超过了可借数量
                alert("您不能再借阅其他图书了!");return;              //弹出提示信息
        }
      form.submit();                                                  //提交表单
    }
</script>
```

技巧 在 JavaScript 中比较两个数值型文本框的值时，不使用运算符“==”，而是将这两个值相减，再判断其结果。

检索读者的基本信息和未归还的借阅图书信息的 SQL 语句如下：

例程22　代码位置：光盘\TM\06\library\bookborrow.php

```
$sql=mysql_query("select r.*,t.name as typename,t.number from tb_reader r left join tb_readerType t on r.typeid=t.id where r.barcode='$barcode'");
$info=mysql_fetch_array($sql);                                        //检索读者信息
```

获取读者借阅信息的 SQL 语句如下：

例程23　代码位置：光盘\TM\06\library\bookborrow.php

```
$sql1=mysql_query("select r.*,borr.borrowTime,borr.backTime,book.bookname,book.price,pub.pubname,bc. name as bookcase from tb_borrow as borr join tb_bookinfo as book on book.id=borr.bookid join tb_publishing as pub on book.ISBN=pub.ISBN join tb_bookcase as bc on book.bookcase=bc.id join tb_reader as r on borr.readerid=r.id   where borr.readerid='$readerid' and borr.ifback=0");
$info1=mysql_fetch_array($sql1);                                      //检索读者的借阅信息
$borrowNumber=mysql_num_rows($sql1);                                  //获取结果集中行的数目
```

在“图书条形码”/“图书名称”文本框中输入图书条形码或图书名称后，单击“确定”按钮，检索图书信息是否存在，如果不存在，则向图书借阅信息表中添加该读者的图书的借阅记录，完成图书借阅操作；否则，弹出该书不能被同一读者重复借阅的提示信息。图书借阅的具体代码如下：

例程24　代码位置：光盘\TM\06\library\bookBorrow.php

```
 <?php
if($_POST[inputkey]!=""){                          //如果“图书条形码”/“图书名称”文本框不为空
$f=$_POST[f];                                      //获取用户选择的条件值
❶   $inputkey=trim($_POST[inputkey]);              //获取用户输入的查询关键字
$barcode=$_POST[barcode];                          //获取读者的条形码
$readerid=$_POST[readerid];                        //获取读者 id 号
❷   $borrowTime=date('Y-m-d');                     //图书的借阅时间为系统当前时间
❸   $backTime=date("Y-m-d",(time()+3600*24*30));   //归还图书日期为当前期日期+30 天期限
$query=mysql_query("select * from tb_bookinfo where $f='$inputkey'");
$result=mysql_fetch_array($query);                 //检索图书信息是否存在
```

```
if($result==false){                                         //如果读者借阅的图书不存在，那么弹出提示信息
    echo "<script language='javascript'>alert('该图书不存在！
');window.location.href='bookBorrow.php?barcode=$barcode'; </script>";
}
else{                                                        //检索该读者所借阅的图书是否与再借图书重复
    $query1=mysql_query("select r.*,borr.borrowTime,borr.backTime,book.bookname,book.price, pub.pubname, bc.
name as bookcase from tb_borrow as borr join tb_reader as r on borr.readerid=r.id join tb_bookinfo as book on
book.id=borr.bookid join tb_publishing as pub on book.ISBN=pub.ISBN join tb_bookcase as bc on book.
bookcase=bc.id where borr.bookid=$result[id] and borr.readerid=$readerid and ifback=0");
    $result1=mysql_fetch_array($query1);
    if($result1==true){                                      //如果所借图书已被该读者借阅，那么提示不能重复借阅
        echo "<script language='javascript'>alert('该图书已经借阅！');window.location.href='bookBorrow.php?
barcode=$barcode';</script>";
    }
    else{                                                    //否则，完成图书借阅操作，并弹出借阅成功提示信息
        $bookid=$result[id];                                 //将读者 id 号赋给一变量
        mysql_query("insert into
tb_borrow(readerid,bookid,borrowTime,backTime,operator,ifback)values('$readerid','$bookid','$borrowTime','$back
Time','$_SESSION[admin_name]',0)");                          //向借阅信息表中添加一条借阅信息
        echo "<script language='javascript'>alert('图书借阅操作成功！'
);window.location.href='bookBorrow.php?barcode=$barcode';</script>";
    }
    }
}
?>
```

代码贴士

❶ trim()：删除字符串中首尾的空白或者其他字符，以达到精确查询。

❷ date('Y-m-d')：获取系统的当前日期为图书借阅的日期，并格式化日期格式。

❸ date("Y-m-d",(time()+3600*24*30))：归还图书日期=当前期日期+30 天期限。当前日期时间戳应用 time()函数获取，30 天期限的时间戳等于 3600 秒 × 24 小时 × 30 天，并通过 date()函数格式化为指定日期格式。

6.8.4　图书续借的实现过程

图书续借使用的数据表：tb_borrow、tb_bookinfo、tb_reader

管理员登录后，选择“图书借还”/“图书续借”菜单项，进入到图书续借页面。在该页面的“读者条形码”文本框中输入读者的条形码（如 123456789）后，单击“确定”按钮，系统会自动检索出该读者的基本信息和未归还的借阅图书信息。如果检索到对应的读者信息，则将其显示在页面中，此时单击“续借”超链接，即可续借指定图书（即将该图书的归还时间加上该书的可借天数 30 天计算得出）。图书续借页面的运行结果如图 6.31 所示。

说明　当读者续借完图书后，操作员通过单击“完成续借”按钮，将重新载入图书续借页面，当前页处于空信息状态，从而方便操作员进行下一个读者续借图书操作。

图书续借页面的设计方法同图书借阅类似，所不同的是，在图书续借页面中没有添加借阅图书的功能，而是添加了“续借”超链接。图书续借页面在 Dreamweaver 中的设计效果如图 6.32 所示。



图 6.31　图书续借页面的运行结果



图 6.32　图书续借页面的设计效果

单击“续借”超链接时，还需要将读者条形码、借阅 id 号和图书归还时间一同传递到图书续借的处理页 borrow_oncemore.php 中。代码如下：

例程25　代码位置：光盘\TM\06\library\bookRenew.php

```
<a href="borrow_oncemore.php?barcode=<?php echo $info[barcode];?>&borrid=<?php echo
$info[borrid];?>&backTime=<?php echo $info[backTime];?>">续借</a>
```

检索读者信息和读者借阅信息的 SQL 语句如下：

例程26　代码位置：光盘\TM\06\library\bookRenew.php

```
$sql=mysql_query("select borr.id as borrid,borr.borrowTime,borr.backTime,borr.ifback,r.*,t.name as typename,
t.number,book.bookname,book.price,pub.pubname,bc.name as bookcase from tb_borrow as borr join tb_reader
r on borr.readerid=r.id join tb_readerType t on r.typeid=t.id join tb_bookinfo as book on book.id=borr.bookid join
tb_publishing as pub on book.ISBN=pub.ISBN  join tb_bookcase as bc on book.bookcase=bc.id where
r.barcode='$barcode' and borr.ifback=0");
$info=mysql_fetch_array($sql);                                         //检索读者信息和借阅信息
```

单击“续借”超链接，提交到数据处理页 borrow_oncemore.php，主要用于完成图书的续借功能，主要通过更改图书的归还日期（即将该图书的归还时间加上该书的可借天数 30 天计算得出，续借日期的具体算法，参见 6.9.1 节的详细讲解）实现。数据处理页的代码如下：

例程27　代码位置：光盘\TM\06\library\borrow_oncemore.php

```
<?php
session_start();                                                       //初始化 session 变量
include("conn/conn.php");                                              //连接数据库文件
$barcode=$_GET[barcode];                                               //获取图书条形码
$new=$_GET[backTime];                                                  //获取图书归还时间
//更新续借期，将动态获取的还书期日转化为时间戳，然后再求出续借后的还书日期
$newbackTime=date("Y-m-d",(mktime(0, 0, 0, substr($new,5,2), substr($new,8,2), substr($new,0,4))+3600*24*30));
$borrid=$_GET[borrid];                                                 //获取续借图书的 id 号
mysql_query("update tb_borrow set backTime='$newbackTime',ifback=0,operator='$_SESSION[admin_name]'
where id=$borrid");
echo "<script language='javascript'>alert('图书续借操作成功！');window.location.href='bookRenew.php? barcode=
$barcode'; </script>";                                                 //弹出图书续借成功的提示信息
?>
```



6.8.5　图书归还的实现过程

图书归还使用的数据表：tb_borrow、tb_bookinfo、tb_reader

管理员登录后，选择“图书借还”/“图书归还”菜单项，进入到图书归还页面。在该页面的“读者条形码”文本框中输入读者的条形码（如 123456789）后，单击“确定”按钮，系统会自动检索出该读者的基本信息和未归还的借阅图书信息。如果检索到对应的读者信息，则将其输出到浏览器，此时单击“归还”超链接，即可将指定图书归还。图书归还页面的运行结果如图 6.33 所示。



图 6.33　图书归还页面的运行结果

图书归还页面的设计方法同图书续借类似，所不同的是，将图书续借页面中的“续借”超链接更换为“归还”超链接。在单击“归还”超链接时，也需要将读者条形码和借阅 id 号一同传递到图书归还处理页。代码如下：

例程28　代码位置：光盘\TM\06\library\bookBack.php

```
<a href="bookBack_ok.php?borrid=<?php echo $info[borrid];?>&barcode=<?php echo $info[barcode];?>">归还
</a>
```

检索读者信息及读者借阅信息的 SQL 语句如下：

例程29　代码位置：光盘\TM\06\library\bookBack.php

```
$sql=mysql_query("select borr.id as borrid,borr.borrowTime,borr.backTime,borr.ifback,r.*,t.name as typename,
t.number,book.bookname,book.price,pub.pubname,bc.name as bookcase from tb_borrow as borr join tb_reader
r on borr.readerid=r.id join tb_readerType t on r.typeid=t.id join tb_bookinfo as book on book.id=borr.bookid join
tb_publishing as pub on book.ISBN=pub.ISBN  join tb_bookcase as bc on book.bookcase=bc.id where
r.barcode='$barcode' and borr.ifback=0");
$info=mysql_fetch_array($sql);                                   //检索读者信息及该读者的借阅信息
```

单击“归还”超链接，即可指定图书归还。数据处理页的代码如下：

例程30　代码位置：光盘\TM\06\library\bookBack_ok.php

```
<?php
session_start();                                                 //初始化 session 变量
include("conn/conn.php");                                        //连接数据库文件
$backTime=date("Y-m-d");                                         //归还图书日期
```

```
$borrid=$_GET[borrid];                                          //获取读者的 id 号
mysql_query("update  tb_borrow  set  backTime='$backTime',ifback=1,operator='$_SESSION[admin_name]'
where id=$borrid");                                             //更新读者的借阅信息
echo "<script language='javascript'>alert('图书归还操作成功！');window.location.href='bookBack.php?barcode=
$barcode';</script>";                                           ///弹出图书归还成功的提示信息
?>
```

6.8.6 图书借阅查询的实现过程

图书借阅查询使用的数据表：tb_borrow、tb_bookinfo、tb_reader

管理员登录后，选择“系统查询”/“图书借阅查询”菜单项，进入到图书借阅查询页面。图书借阅查询页面的运行结果如图 6.34 所示。在该页面中可以按指定的字段或某一时间段进行查询，同时还可以实现按指定字段及时间段进行综合条件查询。



图 6.34 图书借阅查询页面的运行结果

图书借阅查询页面主要用于收集查询条件和显示查询结果，并通过自定义的 JavaScript 函数验证输入的查询条件是否合法。该页面中所涉及的表单元素如表 6.5 所示。

表 6.5 图书借阅查询页面所涉及的表单元素

名 称	元素类型	重要属性	含 义
myform	form	method="post" action=""	表单
flag1	checkbox	value="a"	请选择查询依据
flag2	checkbox	value="b"	借阅时间
f	select	<option value="k.barcode" >图书条形码</option> <option value="k.bookname">图书名称</option> <option value="r.barcode">读者条形码</option> <option value="r.name">读者名称</option>	查询字段
key	text	size="50"	关键字
sdate	text	id="sdate"	开始日期
edate	text	id="edate"	结束日期
Submit	submit	onClick="return check(myform);"	“查询”按钮

在图书借阅查询页面中，指定查询条件后，提交表单信息到当前页。首先获取表单元素复选框 flag 的值，然后根据 flag 的值组合查询字符串。

如果 flag1 的值等于 a，那么按指定的字段检索图书借阅信息；如果 flag2 的值等于 b，那么按指定的时间段检索图书借阅信息；如果 flag1 的值等于 a，并且 flag2 的值等于 b，那么按以上两个条件的综合条件检索图书借阅信息，并将查询结果输出到浏览器。具体代码如下：

例程31　代码位置：光盘\TM\06\library\borrowQuery.php

```
<?php
include("conn/conn.php");                                           //连接数据库文件
$sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id,k.barcode, k.bookname
from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id"); //查询图书借阅信息
if($_POST[Submit]!=""){                                             //如果提交了表单，则执行以下操作
	$f=$_POST[f];                                                   //获取操作员选择的查询条件
	$key1=$_POST[key1];                                             //获取查询关键字
	$sdate=$_POST[sdate];                                           //获取借阅的起始日期
	$edate=$_POST[edate];                                           //获取借阅的结束日期
	$flag1=$_POST[flag1];                                           //获取按指定条件查询的复选框值
	$flag2=$_POST[flag2];                                           //获取按日期查询的复选框值
	if($flag1=="a"){                                                //如果按指定条件查询，则执行以下语句
	$sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id,k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where $f like
'%$key1%'");
	}
	if($flag2=="b"){                                                //如果按时间段查询，则执行以下语句
	$sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name, k.id, k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where
borrowTime between '$sdate' and '$edate'");
	}
	if($flag1=="a" && $flag2=="b"){                                 //如果按综合条件查询，则执行以下语句
	$sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id,
k.barcode,k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id
where borrowTime between '$sdate' and '$edate' and $f like '%$key1%'");
	}
}
$result=mysql_fetch_array($sql);                                    //检索查询结果
if($result==false){                                                 //如果查询结果不存在，则弹出提示信息
?>
<table width="100%" height="30"  border="0" cellpadding="0" cellspacing="0">
  <tr>
    <td height="36" align="center">暂无图书借阅信息！</td>
  </tr>
</table>
<?php
}
else{                                                               //否则，输出图书借阅信息
?>
<table width="723"  border="1" cellpadding="0" cellspacing="0" bordercolor="#FFFFFF" bordercolordark= "#D2E3E6"
bordercolorlight="#FFFFFF">
  <tr align="center" bgcolor="#D0E9F8">
    <td width="13%">图书条形码</td>
```

```
        <td width="27%">图书名称</td>
        <td width="15%">读者条形码</td>
        <td width="11%">读者名称</td>
        <td width="13%">借阅时间</td>
        <td width="11%">归还时间</td>
        <td width="10%">是否归还</td>
    </tr>
<?php
do{
if($result[ifback]=="0"){                                    //如果“是否归还”等于 0，则输出“未归还”
        $ifbackstr="未归还";
}
else{                                                        //如果“是否归还”等于 1，则输出“已归还”
        $ifbackstr="已归还";
}
?>
/* *********************************输出符合查询条件的记录***************************** */
<tr>
    <td style="padding:5px;"> <?php echo $result[barcode];?></td>
    <td style="padding:5px;"><a href="book_look.php?id=<?php echo $result[id]; ?>"><?php echo $result
[bookname]; ?></a></td>
    <td style="padding:5px;"> <?php echo $result[readerbarcode];?></td>
    <td style="padding:5px;"> <?php echo $result[name];?></td>
    <td style="padding:5px;"> <?php echo $result[borrowTime];?></td>
    <td style="padding:5px;"> <?php echo $result[backTime];?></td>
    <td style="padding:5px;"> <?php echo $ifbackstr;?></td>
</tr>
/* ************************************************************************************** */
<?php
        }while($result=mysql_fetch_array($sql));
}
?>
```

6.8.7 单元测试

在开发完图书借还模块后，对该模块进行了单元测试。当操作员按指定字段进行条件查询时，能够实现图书借阅查询；当操作员按指定的时间段进行查询时，也能够实现图书借阅查询；但是当操作员按以上两个条件同时进行查询时，则不能实现图书借阅查询。

图书借还模块的表单元素代码如下：

```
<input name="flag" type="checkbox" class="noborder" value="a" checked>
<input name="flag" type="checkbox" class="noborder" id="flag" value="b">
```

完成图书借阅查询的 SQL 语句如下：

```
if($flag=="a"){
        $sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id, k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where $f like
'%$key1%'");
```



```
}
if($flag=="b"){
    $sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id, k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where
borrowTime between '$sdate' and '$edate'");
}
if($flag=="a" && $flag=="b"){
    $sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id,k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where
borrowTime between '$sdate' and '$edate' and $f like '%$key1%'");
}
```

为了找出错误的原因，笔者应用 echo()语句对$flag 的值进行输出，输出的结果显示$flag 等于 b，不执行$flag 等于 a。这充分说明在对同名复选框进行查询时，它只能获取最后一个复选框的值。

解决该问题的方法是将两个复选框分别定义为两个不同的名称，然后赋予不同的值，再进行判断，就可以实现按指定字段、指定的条件进行综合条件查询了。

因此将表单中的 HTML 标记中的 flag 进行重命名，代码如下：

```
<input name="flag1" type="checkbox" class="noborder" value="a" checked>
<input name="flag2" type="checkbox" class="noborder" id="flag" value="b">
```

修改后的完成图书借阅查询的 SQL 语句如下：

```
if($flag1=="a"){
    $sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id,k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where $f
like '%$key1%'");
}
if($flag2=="b"){
    $sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id,k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where
borrowTime between '$sdate' and '$edate'");
}
if($flag1=="a" && $flag2=="b"){
    $sql=mysql_query("select b.borrowTime,b.backTime,b.ifback,r.barcode as readerbarcode,r.name,k.id,k.barcode,
k.bookname from tb_borrow b join tb_reader r on b.readerid=r.id join tb_bookinfo k on b.bookid=k.id where
borrowTime between '$sdate' and '$edate' and $f like '%$key1%'");
}
```

6.9　开发技巧与难点分析

6.9.1　如何自动计算图书归还日期

在图书馆管理系统中会遇到这样的问题：在借阅图书时，需要自动计算图书的归还日期。

1．图书归还日期

根据图书馆还书的规律一般都以 30 天为一个期限，因此在图书归还时，可以设置一个固定的值，即 30 天。计算归还日期的方法如下：

图书归还日期=系统当前日期+借阅天数固定值 30 天。

自动计算图书归还日期的具体代码如下：

```
date("Y-m-d",(time()+3600*24*30))                              //图书归还日期
```

2．续借图书归还日期

续借图书归还日期是在原来数据库保存该图书归还日期（这个日期是不固定的）的基础上再次借阅所计算的时间，它是需要根据数据表中保存的归还日期来计算的。计算图书续借归还日期的方法如下：

续借图书归还日期=所借图书在数据表中的归还日期+借阅天数固定值 30 天。

首先应用 substr()函数分别取出所借图书在数据表中原定的归还日期“月”、“日”、“年”，然后应用 mktime()函数计算出归还日期的时间戳，最后应用 date()函数格式化日期为“YYYY-MM-DD”格式。自动计算续借图书归还日期的代码如下：

```
$new=$_GET[backTime];                                    //获取传递过来的该图书在数据表中的归还日期
//更新续借期，将动态获取的还书日期转化为时间戳，然后再求出续借后的还书日期
date("Y-m-d",(mktime(0, 0, 0, substr($new,5,2), substr($new,8,2), substr($new,0,4))+3600*24*30));
```

6.9.2 如何对图书借阅信息进行统计排行

在图书馆管理系统的首页中，提供了显示图书借阅排行榜功能。要实现该功能，最重要的是如何获取统计排行信息，这可以通过一条 SQL 语句实现。本系统中实现对图书借阅信息进行统计排行的 SQL 语句如下：

```
select * from (select bookid,count(bookid) as degree from tb_borrow group by bookid) as borr join (select b.*,
c.name as bookcasename,p.pubname,t.typename from tb_bookinfo b left join tb_bookcase c on b.bookcase=
c.id join tb_publishing p on b.ISBN=p.ISBN join tb_booktype t on b.typeid=t.id where b.del=0) as book on borr.
bookid=book.id order by borr.degree desc limit 10
```

下面将对该 SQL 语句进行分析：

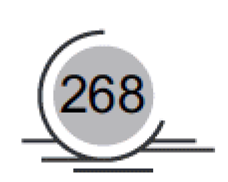
（1）对图书借阅信息表进行分组并统计每本图书的借阅次数，然后使用 as 为其指定别名为 borr。代码如下：

```
(select bookid,count(bookid) as degree from tb_borrow group by bookid) as borr
```

（2）使用左联接查询出图书的完整信息，然后使用 as 为其指定别名为 book。代码如下：

```
(select b.*,c.name as bookcasename,p.pubname,t.typename from tb_bookinfo b left join tb_bookcase c on
b.bookcase=c.id join tb_publishing p on b.ISBN=p.ISBN join tb_booktype t on b.typeid=t.id where b.del=0) as book
```

（3）使用 join on 语句将 borr 和 book 连接起来，再对其按统计的借阅次数 degree 进行降序排序，



并使用 limit 子句限制返回的行数。

6.10　联接语句技术专题

在实际网站开发过程中，经常需要从多个表中查询信息，在 MySQL 数据库中可以通过连接的方式实现多表查询，连接方式分为内联接和外联接两种。下面对这两种连接方式进行详细讲解。

6.10.1　内联接语句

inner join 即内联接查询方式，是程序开发中常用的连接方式。内联接称为相等联接，它返回两个表中的所有列，但只返回在联接列中具有相等值的行。内联接查询的语法格式如下：

```
select fieldlist
from   table1 [inner] join table2
on table1.column=table2.column
```

参数说明如下。

- ☑ fieldlist：要查询的字段列表。
- ☑ table1、table2：为要连接的表名。
- ☑ inner：可选项，表示表之间的连接方式为内联接。
- ☑ on table1.column=table2.column：用于指明表 table1 和表 table2 之间的连接条件。

下面通过内联接方式实现员工信息表和员工工资表的连接，并显示查询结果。代码如下：

```
$sql=mysql_query("select tb_yg.userid,tb_yg.name,tb_yg.sex,tb_yg.age,tb_yg.tel,tb_yg.bm,tb_yg_info.gz from
tb_yg inner join tb_yg_info on tb_yg.userid=tb_yg_info.ygid");
$info=mysql_fetch_array($sql);
```

6.10.2　外联接语句

内联接返回的是两个表中符合条件的数据，而外联接返回部分或全部匹配行，这主要取决于所建立的外联接的类型。外联接分为左外联接和右外联接，下面对这两个外联接的使用方法进行详细讲解。

1．左外联接（left outer join）

左外联接返回的查询结果包含左表中的所有符合查询条件及右表中所有满足连接条件的行。MySQL 数据库中使用左外联接的语法格式如下：

```
select field 1[field2…]
     from table1 left [outer] join table2
     on join_condition
[where search_condition]
```

参数说明如下。

- ☑ left outer join：表示表之间通过左外联接方式相互连接，也可以简写成 left join。

☑ on join_condition：指多表建立连接所使用的连接条件。
☑ where search_condition：可选项，用于设置查询条件。

下面通过左外联接的方式建立员工信息表和员工工资表的连接，并显示查询结果。代码如下：

```
$sql=mysql_query("select * from tb_yg left outer join tb_yg_info on tb_yg.userid=tb_yg_info.ygid ",$conn);
$info=mysql_fetch_array($sql);
```

2．右外联接（right outer join）

右外联接返回的查询结果包含左表中的所有符合连接条件以及右表中所有满足查询条件的行。MySQL 数据库中使用右外联接的语法格式如下：

```
select field 1[field2…] from table1 right [outer] join table2 on join_condition [where search_condition]
```

参数说明如下。

☑ right outer join：表示表之间通过右外联接方式相互连接，也可以简写成 right join。
☑ outer：可选项，表示表之间的联接方式为完全联接。
☑ on join_condition：指多表建立联接所使用的连接条件。
☑ where search_condition：可选项，用于设置查询条件。

下面通过右外联接的方式建立员工信息表和员工工资表的连接，并显示查询结果。代码如下：

```
$sql=mysql_query("select * from tb_yg right outer join tb_yg_info on tb_yg.userid=tb_yg_info.ygid ");
$info=mysql_fetch_array($sql);
```

6.11 本章总结

本章运用软件工程的设计思想，通过一个完整的图书馆管理系统引导读者深入了解系统的开发流程。在这个系统的实现过程中，除了应用一些基本的 PHP 技术之外，还涉及一些独特的技术细节，如权限设置、多表查询技术等需要读者掌握，并能在实际的操作中灵活应用，举一反三。

第 7 章

博客管理系统

（Apache+PHP+phpMyAdmin+MySQL 5.0 实现）

如果有人问“什么是 Web 2.0？”，恐怕没几个人能说得清楚，但是要问什么是博客，哪怕是从不上网的人也是耳熟能详的了。博客正是 Web 2.0 概念中重要的组成部分之一（大家熟知的还包括 IM 即时通和 RSS 阅读器）。

Blog（博客），全名 Weblog，后来缩写为 Blog。Blogger 就是写 Blog 的人，习惯于在网上写出日记、发布个人照片、展示个性自我的用户群体。对于 Blog/Blogger 的中文名称，有翻译成“博客”，也有翻译为“网志”，但大多数人都已经认可了“博客”。

通过阅读本章，可以学习到：

- **博客管理系统的开发流程**
- **进一步掌握如何做项目需求分析与系统设计**
- **实现一个简单的公告栏管理模块**
- **掌握不同的图片上传技术**
- **掌握一种采用 JavaScript+CSS 技术来实现的半透明动态下拉菜单**

7.1 开发背景

博客管理系统给人们提供抒发个人情感、人与人之间进行良好沟通的平台，博客拥有真实的内容，可以通过博客记录工作、学习、生活和娱乐的点滴，以及发表文章和评论，从而在网上建立一个完全属于自己的个人天地，成为当今网络最为个性化和平民化的个人展示空间。对于网民，只要拥有博客，就可以超越现实生活，拥有不同以往的全新网上生活；对于博客服务商，则必须从功能提供转型到全方位社会服务的提供，建立虚拟社会，并负责维护运行，保证博客日常生活；对于社会而言，有利于构建和谐的互联网空间，维护和谐的社会环境。从这个角度来说，构建新生活方式，将是互联网发展的一个里程碑。

7.2 需求分析

信息时代的今天，博客已经成为一种新的生活方式。在网络中构建一个赋有个性化的个人博客，提供了一种可信任的和实时连通的网络环境，通过网络开放性和交互性的特点，让用户在任何时间、任何地点，通过网络方便地“生活”，不仅是信息传递与获取，还可以进行群体交流和资源共享，展示自我，为个人发展带来新机遇。

通过对多个博客网的调查分析，客户要求本博客管理系统具有以下功能：

☑ 要求系统采用 B/S 架构，实现人机交互。
☑ 要求系统界面个性化，色彩搭配和谐，具有很强的视觉冲击力，操作简便。
☑ 要求突出主题，显示最新文章和公告。
☑ 要求游客可以浏览文章、浏览图片、发表评论。
☑ 要求具有强大的搜索查询功能，实现精确查询和模糊查询。
☑ 完善的文章管理功能，包括文章的发表、删除，以及对文章的评论与回复。
☑ 支持图片上传功能，可以上传各种类型的图片。
☑ 支持好友功能。
☑ 系统运行稳定，安全可靠。

7.3 系统设计

7.3.1 系统目标

该系统主要实现如下目标：

☑ 系统采用 B/S 架构，实现人机交互。
☑ 系统界面设计以浅色为主，美观友好，操作简便。
☑ 突出重点内容，显示最新文章。
☑ 非登录用户可以浏览文章、浏览图片、发表评论。



☑ 全面的搜索查询功能，包括精确查询和模糊查询。
☑ 完善的文章管理功能，包括文章的发表、删除，以及对文章的评论与回复。
☑ 支持图片上传功能。
☑ 支持好友功能。
☑ 支持公告栏功能。
☑ 系统运行稳定，安全可靠。

7.3.2　系统功能结构

博客管理系统的功能结构如图 7.1 所示。

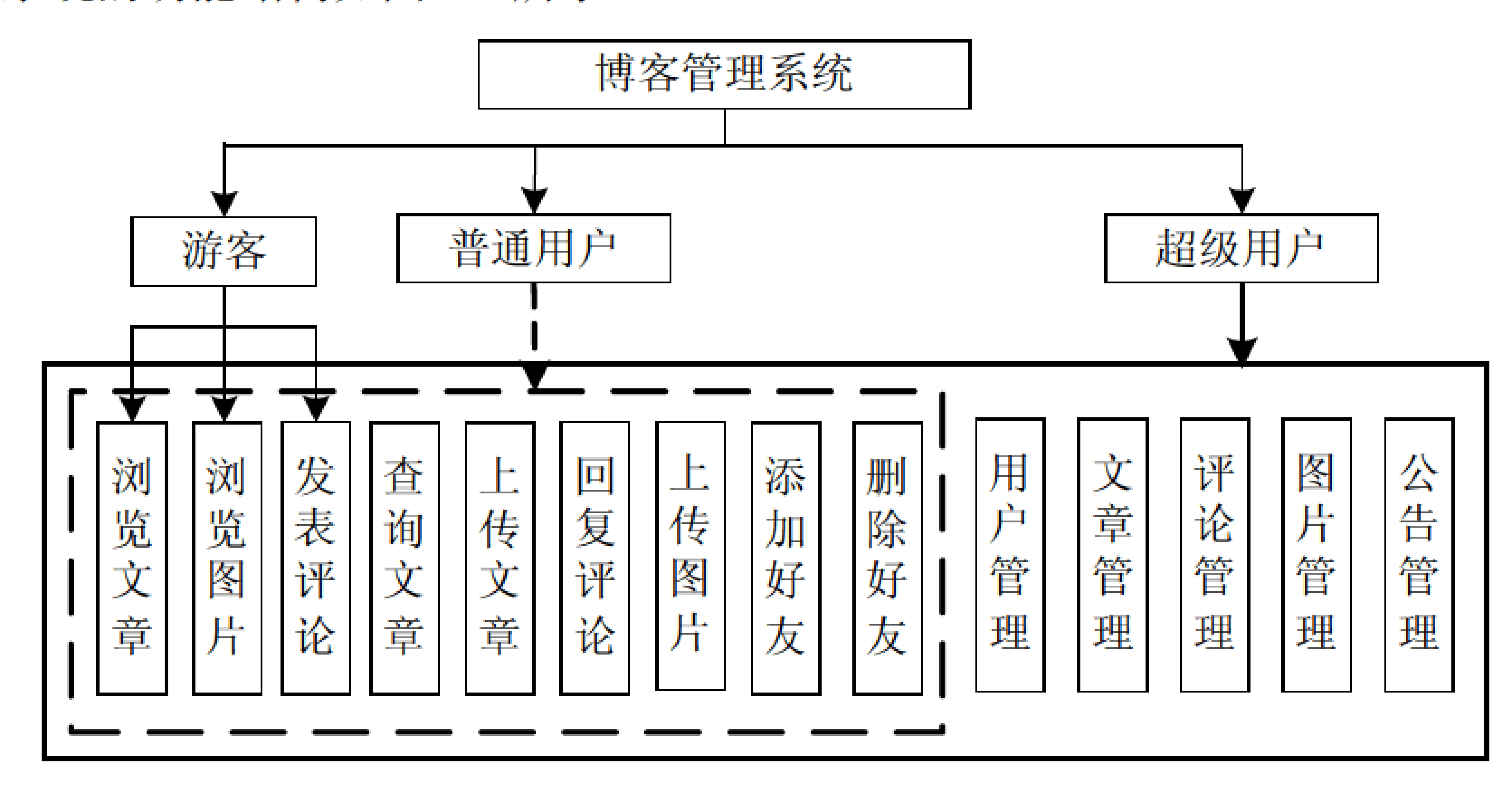


图 7.1　博客管理系统功能结构图

7.3.3　系统功能预览

为了让读者对本系统有个初步的了解和认识，下面给出本系统的几个页面运行效果图，如果想查看完整的效果图，请参见光盘源程序。

博客管理系统的首页如图 7.2 所示，该页面包含了系统大部分的功能链接，包括用户注册、用户登录、文章浏览等。用户注册页面如图 7.3 所示，该页面显示了用户注册时需要填写的资料、注意事项等。



图 7.2　博客首页　（光盘\TM\07\online\index.php）



图 7.3　用户注册（光盘\TM\07\online\Register.php）

浏览文章页面如图 7.4 所示，该页面用于显示文章及相关的评论，也可以发表评论。发表文章页面如图 7.5 所示，该页面用于登录用户发表文章，包括文章标题、文字编辑区和文章内容。



图 7.4　浏览文章（光盘\TM\07\online\article.php）　　图 7.5　发表文章（光盘\TM\07\online\file.php）

图片上传页面用于上传图片，如图 7.6 所示，通过该页面用户可以将图片或照片添加到博客中。添加好友页面如图 7.7 所示，该页面用于输入用户好友的详细信息，包括姓名、性别、生日等。





图 7.6　图片上传（光盘\TM\07\online\add_pic.php）　　图 7.7　添加好友（光盘\TM\07\online\friendd.php）

7.3.4　系统流程图

博客管理系统的流程图如图 7.8 所示。

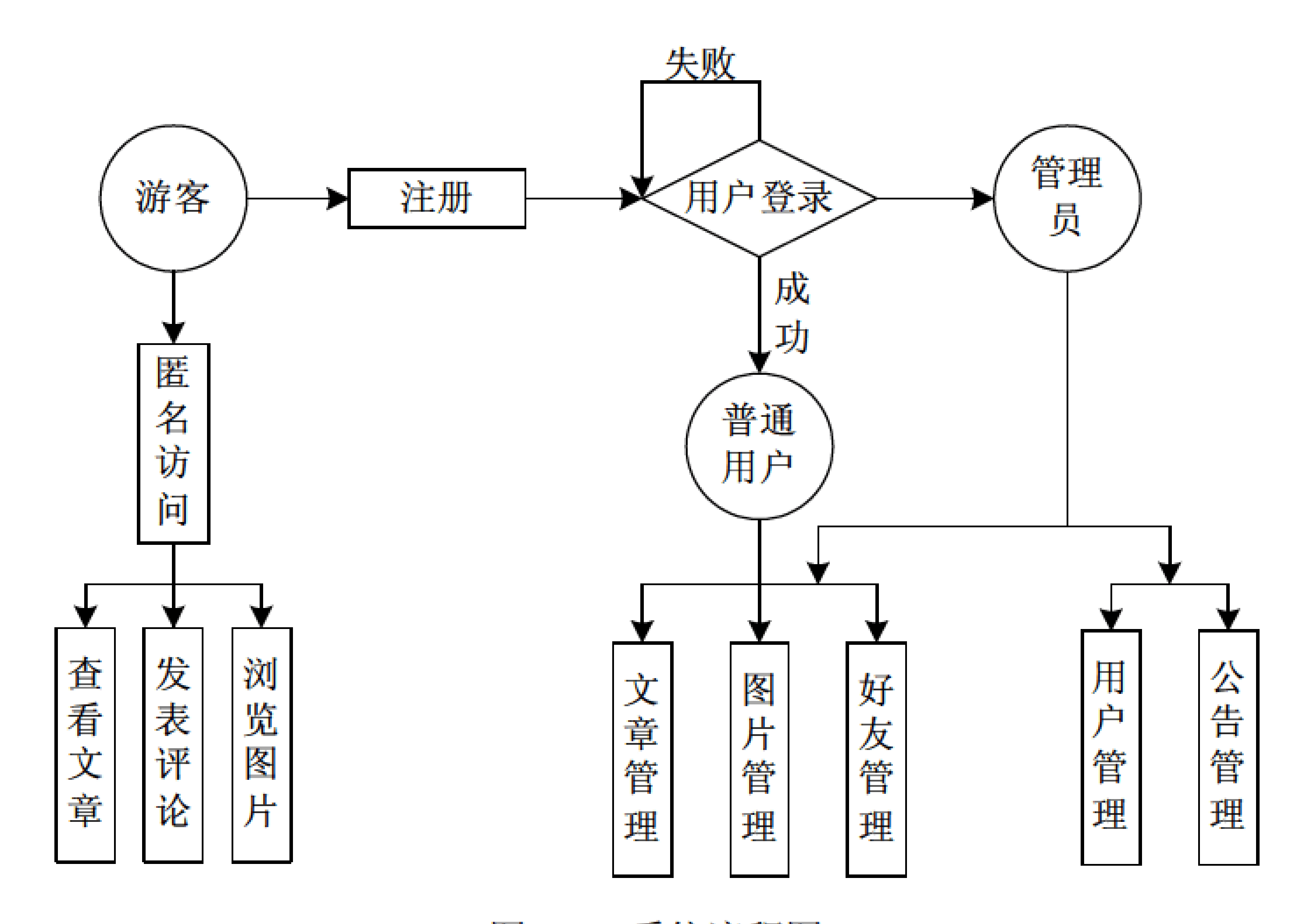


图 7.8　系统流程图



7.3.5　开发环境

在开发博客管理系统平台时，该项目使用的软件开发环境如下。

1．服务器端

- ☑ 操作系统：Windows Server 2003。
- ☑ 服务器：Apache 2.2.8。
- ☑ PHP 软件：PHP 5.2.6。
- ☑ 数据库：MySQL 5.0.51。
- ☑ MySQL 图形化管理软件：phpMyAdmin-2.10.3。
- ☑ 开发工具：Dreamweaver 8。
- ☑ 浏览器：IE 6.0。
- ☑ 分辨率：最佳效果为 1024×768 像素。

2．客户端

- ☑ 浏览器：推荐使用 IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

7.3.6　文件夹组织结构

博客系统的目录比较少，结构比较简单，主要有数据库链接文件目录、css 样式目录、JS 脚本目录及背景图片目录。文件夹组织结构如图 7.9 所示。

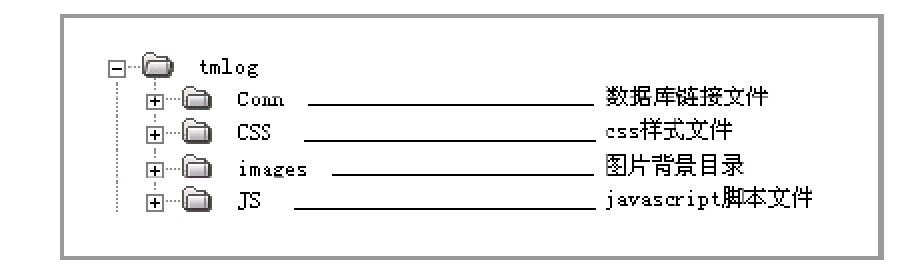


图 7.9　博客管理系统文件夹组织结构

7.4　数据库设计

7.4.1　数据库分析

本系统属于中小型个人网站，毫无争议的，本系统采用的依然是 PHP+MySQL 这对黄金组合，无论是从成本、性能、安全上考虑，还是从易操作性上考虑，MySQL 都是最佳选择。

7.4.2　数据库概念设计

通过需求分析和功能上的设计，本系统规划出用户信息实体、上传图片实体、朋友圈实体、文章实体和留言实体。下面给出主要的实体及 E-R 图。

用户信息实体包括注册用户的详细个人信息，如果想在本系统中进行发表文章、上传图片等操作，则必须要先进行注册。用户信息实体 E-R 图如图 7.10 所示。

上传图片实体主要包括图片名称、图片 id、上传用户、上传空间和图片空间等，实体 E-R 图如图 7.11

所示。

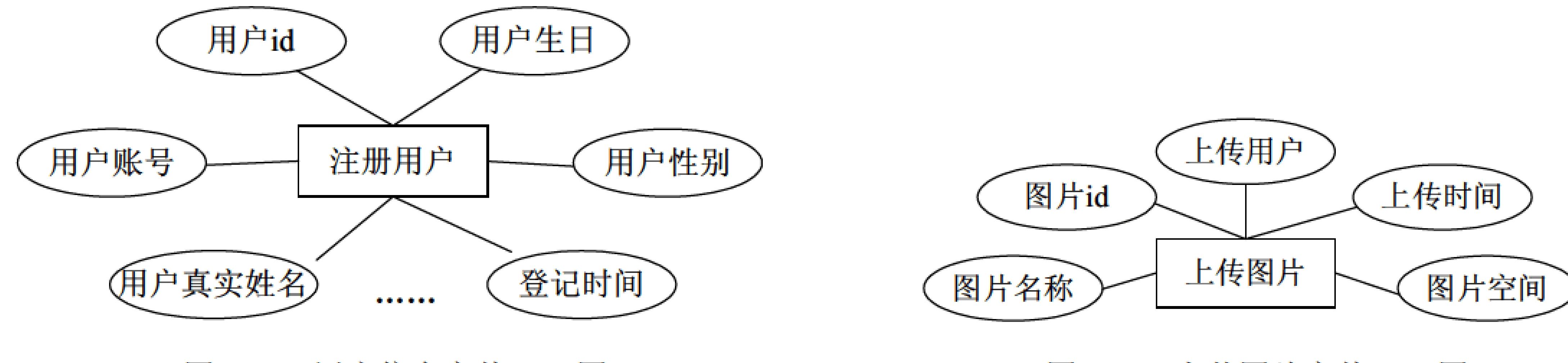


图 7.10　用户信息实体 E-R 图　　　　图 7.11　上传图片实体 E-R 图

7.4.3　数据库物理结构设计

根据实体 E-R 图和本系统的实际情况，需要创建 6 张数据表，如图 7.12 所示。

下面来具体看一下这 6 张数据表的结构设计。

服务器: localhost ▸ 数据库: db_tmlog

表	类型	整理	说明
tb_article	MyISAM	gb2312_chinese_ci	文章列表
tb_filecomment	MyISAM	gb2312_chinese_ci	评论列表
tb_friend	MyISAM	gb2312_chinese_ci	好友列表
tb_public	MyISAM	gb2312_chinese_ci	公告列表
tb_tpsc	MyISAM	gb2312_chinese_ci	图片列表
tb_user	MyISAM	gb2312_chinese_ci	用户列表

图 7.12　数据表列表

1．tb_user（用户列表）

用户列表主要存储用户的个人信息。tb_user 表的结构如图 7.13 所示。

2．tb_article（文章列表）

文章列表存储的是用户发表过的文章信息。tb_article 表的结构如图 7.14 所示。

服务器: localhost ▸ 数据库: db_tmlog ▸ 表 : tb_user

字段	类型	整理	属性	Null	默认	说明
id	int(20)			否		自动编号
regname	varchar(20)	gb2312_chinese_ci		否		用户账号
regrealname	varchar(20)	gb2312_chinese_ci		否		真实姓名
regpwd	varchar(40)	gb2312_chinese_ci		否		用户密码
regbirthday	date			否	0000-00-00	用户生日
regemail	varchar(100)	gb2312_chinese_ci		否		E-mail
regcity	varchar(100)	gb2312_chinese_ci		否		所在城市
regico	varchar(50)	gb2312_chinese_ci		否		用户头像
regsex	varchar(4)	gb2312_chinese_ci		否		用户性别
regqq	varchar(40)	gb2312_chinese_ci		否		用户qq
reghomepage	varchar(100)	gb2312_chinese_ci		否		用户主页
regsign	varchar(200)	gb2312_chinese_ci		否		个人签名
regintroduce	mediumtext	gb2312_chinese_ci		否		自我介绍
ip	varchar(20)	gb2312_chinese_ci		否		登录IP
flg	int(1)			否	0	管理员

图 7.13　用户列表结构

服务器: localhost ▸ 数据库: db_tmlog ▸ 表 : tb_article

字段	类型	整理	属性	Null	默认	说明
id	int(10)			否		自动编号
title	varchar(200)	gb2312_chinese_ci		否		文章标题
content	mediumtext	gb2312_chinese_ci		否		文章内容
author	varchar(20)	gb2312_chinese_ci		否		文章作者
now	datetime			否	0000-00-00 00:00:00	上传时间

图 7.14　文章列表结构

3．tb_filecomment（评论列表）

评论列表存储的是用户对文章的评论，包括注册用户和游客都可以发表评论。tb_filecomment 表的结构如图 7.15 所示。

4．tb_tpsc（图片列表）

图片列表存储的是上传图片的信息，如图片名称、上传用户、上传时间等。tb_tpsc 表的结构如图 7.16 所示。

服务器: localhost ▸ 数据库: db_tmlog ▸ 表 : tb_filecomment

字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
fileid	int(4)			否	0	文章id
username	varchar(20)	gb2312_chinese_ci		否		评论用户
content	text	gb2312_chinese_ci		否		评论内容
datetime	datetime			否	0000-00-00 00:00:00	评论时间

图 7.15　评论列表结构

服务器: localhost ▸ 数据库: db_tmlog ▸ 表 : tb_tpsc

字段	类型	整理	属性	Null	默认	说明
id	int(10)			否		自动编号
tpmc	varchar(30)	gb2312_chinese_ci		否		图片名称
file	mediumblob		BINARY	否		二进制图片
author	varchar(20)	gb2312_chinese_ci		否		上传用户
scsj	date			否	0000-00-00	上传时间

图 7.16　图片列表结构

5．tb_friend（好友列表）

好友列表主要记录了姓名、性别、生日等好友的个人信息。tb_friend 表的结构如图 7.17 所示。

6．tb_public（公告列表）

公告列表主要记录了网站情况、博客系统的版本情况或是网站活动等。公告列表的结构如图 7.18 所示。

服务器: localhost ▸ 数据库: db_tmlog ▸ 表 : tb_friend

字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
name	varchar(50)	gb2312_chinese_ci		否		好友姓名
sex	varchar(10)	gb2312_chinese_ci		否		好友性别
bir	date			否	0000-00-00	好友生日
city	varchar(50)	gb2312_chinese_ci		否		所在城市
address	varchar(100)	gb2312_chinese_ci		否		家庭住址
postcode	varchar(6)	gb2312_chinese_ci		否		邮政编码
email	varchar(50)	gb2312_chinese_ci		否		E-mail
tel	varchar(20)	gb2312_chinese_ci		否		座机号码
handset	varchar(20)	gb2312_chinese_ci		否	0	手机号码
QQ	varchar(20)	gb2312_chinese_ci		否	0	好友QQ
username	varchar(20)	gb2312_chinese_ci		否		用户昵称

图 7.17　好友列表结构



服务器: localhost ▸ 数据库: db_tmlog ▸ 表 : tb_public

字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
title	varchar(50)	gb2312_chinese_ci		否		公告主题
content	varchar(200)	gb2312_chinese_ci		否		公告内容
pub_time	date			否		发布时间

图 7.18　公告列表结构

7.5　首　页　设　计

7.5.1　首页概述

本系统首页页面设计简洁，主要包括以下 3 部分内容。

☑　首部导航栏：包括首页链接、注册和登录模块。

☑　左侧显示区：包括最新文章、最新图片和系统时间模块。游客主要通过该区域浏览文章、浏览图片及发表评论。

☑　主显示区：为系统公告栏，显示系统及网站的最新咨询。

本案例中提供的首页如图 7.19 所示。该首页在本书光盘中的路径为\TM\07\tmlog\index.php。

7.5.2　首页技术分析

在首页主显示区，是一个公告栏模块。公告栏主要用于公布系统版本的更新或升级情况、网站的最新活动安排等，也可以链接一些用户的精彩文章。本系统的公告栏模块是通过<marquee>标签来实现的。<marquee>标签是 HTML 自带的，也是初学者最常用的公告栏实现方式。使用<marquee>标签，可

以实现文字或图片的滚动效果，增加了实用性的同时，也增加了观赏性。下面就来学习一下这个标签的常用方法及属性。

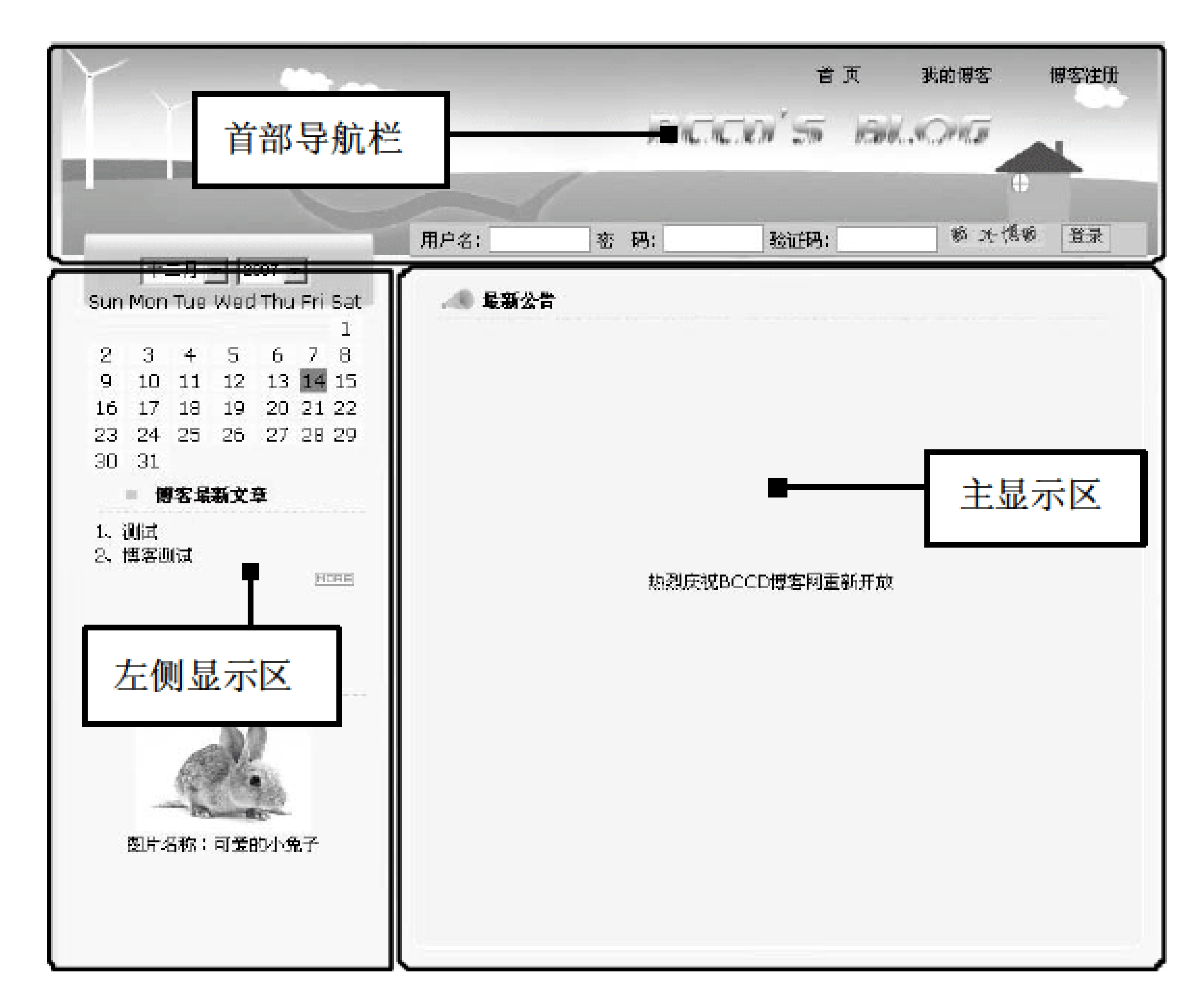


图 7.19　博客管理系统首页

注意 <marquee>标签是微软与网景等公司私制的，从来没有被 W3C 当作正式标签来使用，所以除了 IE 浏览器外，有些属性是不被其他浏览器所支持的，在使用时一定要留意。

1. <marquee>标签的文字移动属性

<marquee>标签的特点就是可以使文字或图片动起来，在早些时候，这可是一个了不起的技术。随着 W3C 标准的逐渐完善，<marquee>标签已经越来越少有用武之地了，但有些技术是永远不变的，如<marquee>标签中的属性名称和属性值，了解 css 样式表和 JavaScript 脚本语言的人会感觉到非常熟悉，因为这些属性名称和属性值都经常被提及和使用。<marquee>标签用的文字移动属性及说明如表 7.1 所示。

表 7.1　<marquee>标签常用的文字移动属性及说明

属性名称	属性值	属性说明	应用举例
marquee	无	标签基本语法，除了文字外，还可以是图片	<marquee>你好，PHP</marquee>
direction	left,right,up,down	文字移动属性，分别表示从右往左、从左往右、从下到上、从上到下	<marquee direction="up">从下到上移动</marquee>
behavior	scroll,slide,alternate	文字移动方式，分别表示沿同一方向不停滚动、只滚动一次、在两个边界内来回滚动	<marquee behavior="scroll">不停的循环播放</marquee>
loop	数值 1,2,3…	循环次数，不指定则表示为无限循环	<marquee loop=5 behavior=slide>只循环滚动 3 次</marquee>
scrollmount	数值 10,20,30…	滚动速度，数值越大，速度越快	<marquee scrollmount=25>我漂起来了！!</marquee>
Scrolldelay	数值 100,200,300…	延时，每动一次，停止的时间	<marquee scrolldelay="500" scrollamount=100>停停走走</marquee>



2．<marquee>标签的外观设置

<marquee>标签不仅可以设置文字移动效果，还能设置标签的外观。外观包括对齐方式、标签底色、面积等。<marquee>标签常用的外观属性及说明如表 7.2 所示。

表 7.2　<marquee>标签常用的外观属性及说明

属性名称	属　性　值	属性说明	应用举例
align	top,middle,bottom	指定文本的对齐方式	<marquee align="middle">我在中间</marquee>
bgcolor	颜色值"#000000"	滚动栏的颜色	<marquee bgcolor="#DEEBEF">看到了吗?</marquee>
height	数值 10,20,30…	滚动栏高度	<marquee height="15">不要，太矮了</marquee>
width	数值 10,20,30…	滚动栏宽度	<marquee width="100">哇！好大的房子啊</marquee>

3．<marquee>标签的其他属性

除了以上的常规设置，<marquee>标签还有两个特殊的属性，即 start()和 stop()，它们可以配合 JavaScript 事件产生很有意思的效果。例如：

```
onMouseOut="this.start()";                              //当鼠标移出该区域时，开始滚动
onMouseOver="this.stop()";                              //当鼠标进入该区域时，停止滚动
```

7.5.3　首页的实现过程

博客管理系统采用二分栏结构，表单布局。具体实现代码如下：

例程01　代码位置：光盘\TM\07\tmlog\index.php

```
<?php
session_start();                                        //开启 session 支持
include "Conn/conn.php";                                //包含数据库链接文件
?>
<table width="757"   border="0" align="center" cellpadding="0" cellspacing="0">
  <tr align="right" valign="top">
    <td height="149" colspan="2" background="images/head.jpg">
     …                                    <!--上部导航栏-->
  </td>
  </tr>
  <tr>
    <td width="236" height="501" background=" images/left.jpg">
     …                                    <!--左侧导航栏-->
  </td>
    <td width="521" height="501" align="center" background=" images/right.jpg">
         <!-- -------------------------------系统公告区------------------------------------ -- >
              <?php
                   $p_sql = "select * from tb_public order by id desc";   //从数据表中读取数据
                   $p_rst = mysql_query($p_sql,$link);                  //执行 SQL 语句
              ?>
              <! -- -----------------------------<marquee>标签开始------------------------------ -- >
```

```
❶                <marquee onMouseOver=this.stop() style="WIDTH: 426px; HEIGHT: 280px" onMouseOut=this.
start()   scrollamount=2 scrolldelay=7 direction=up>
                <?php
                    while($p_row = mysql_fetch_row($p_rst)){                         //循环输出公告标题
                ?>
❷                   <a href="#" onclick="wopen=open('show_pub.php?id=<?php echo
$p_row[0]; ?>','','height=200,width=500,scollbars=no')"><?php echo $p_row[1]; ?></a><br>
                <?}?>
            </marquee>
            <! -- ---------------------------------------------------------------------------------------------------- -->
        </td>
    </tr>
</table>
```

代码贴士

❶ <marquee>标签：<marquee>标签中包含了 start 和 stop 两个特殊的属性，还有速度属性 scollamount 和延迟属性 scrolldelay，而公告栏的高度（height）和宽度（width）则是通过 css 样式设置的。

❷ <?php echo $p_row[1]; ?>：循环输出公告标题，并为每个公告添加超链接，这里用到了 JavaScript 脚本的 open()方法，该方法的作用是打开一个新窗口，同时可以对窗口样式做出设置。open()方法的一般格式为：open("链接的 url","自定义名称","窗口样式")。

7.6 文章管理模块设计

7.6.1 文章管理模块概述

对一个博客系统来说，文章管理是最基本的功能，但同时也是最复杂的一个功能。本系统的文章管理模块包括“添加博客文章”、“查找博客文章”、“管理我的博客”、“发表评论”、“删除文章”和“删除评论”等 6 大功能。其中，普通用户只能删除自己的文章及对文章的评论，只有管理员才有权删除任何一篇文章及回复。文章管理模块的框架如图 7.20 所示。

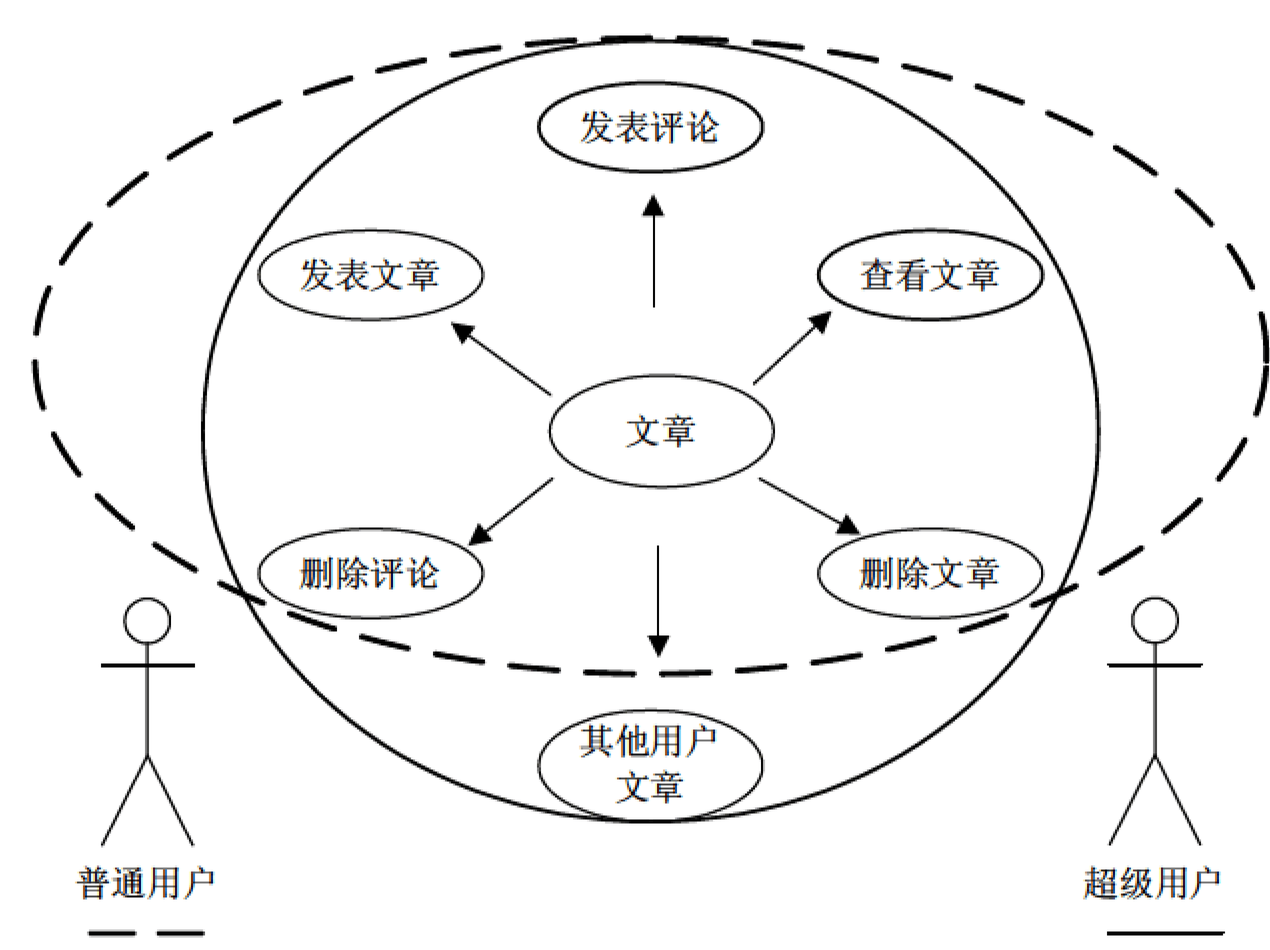


图 7.20　文章管理模块框架图



7.6.2　文章管理模块技术分析

想要使用文章管理模块，前提是用户必须登录，匿名用户是无法访问这些功能的；想要删除文章和评论，前提是当前用户要么是管理员权限，要么是文章拥有者，否则不会显示删除功能的。这两方面的控制都需要 session 的配合，本节就来讲一下 session 的应用及常见的问题处理。

session 的中文译名为“会话”，是指用户从进入网站开始，直到关闭网站这段时间内，所有网页共同使用的公共变量的存储机制。session 比 cookie 更有优势：session 是存储在服务器端的，不易被伪造；session 的存储没有长度限制；session 的控制更容易等。

PHP 中的 session 功能一直令许多初学者望而生畏，因为经常出现一些莫名其妙的错误，而又不知道如何去解决。其实，大多数的错误都是因为对 session 的配置不了解而使用方法不正确造成的，在 php.ini 中对 session 的配置如表 7.3 所示。

表 7.3　session 的常用配置选项

配 置 选 项	说　明
session.save_path = c:/temp	保存 session 变量的目录，在 Linux/UNIX 下为/tmp
session.ues_cookies = 1	是否使用 cookie
session.name = PHPSESSID	表示会话 ID
session.auto_start = 0	是否自动启用 session，当为 1 时，在每页中就不必调用 session_start()函数了
session.cookie_lifetime = 0	设定 cookie 送到浏览器后的保存时间，单位为秒。默认值为 0，表示直到浏览器关闭
session.cookie_path = /	cookie 有效路径
session.cookie_domain =	有效域名
session.serialize_handler = php	定义序列化数据的标识，本功能只有 WDDX 模块或 PHP 内部使用，默认值为 PHP
session.gc_probability = 1	设定每次临时文件开始处理的处理概率。默认值为 1
session.gc_maxlifetime = 1440	设定保存 session 的临时文件被清除前的存活秒数
session.referer_check =	决定参照到客户端的 session 代码是否要删除。有时出于安全或其他考虑，会设定不删除。默认值为 0
session.cache_limiter = nocache	设定 session 缓冲限制
session.cache_expire = 180	文档有效期，单位为分钟
session.save_handler = files	用于保存 session 变量，默认情况下用文件

对于初学者来说，session 在 php.ini 中不需要特意去改动，因为安装时会根据操作系统自行做出适当的调整。只在少数的几项，如 session 存活周期（session.cookie_lifetime = 0）、自动开启 session（session.auto_start）等稍加改动即可。

PHP 主要是通过会话（session）处理函数来对 session 进行控制和使用的。常用的处理函数如表 7.4 所示。

表 7.4　PHP 常用的会话处理函数

函　　数	函 数 说 明
session_start();	开启 session 或返回已经存在的 session
$_SESSION['name'] = value;	注册一个 session 变量
session_id()	设定或取得当前的 session_id 值

续表

函　　数	函 数 说 明
isset($_SESSION['name'])	检测指定的 session 值是否存在。isset 不只可以检测 session，还可以检测其他类型，如 isset($_POST['name'])、isset($_GET['name'])等
session_regenerate_id()	更改 session_id 的值
session_name()	返回或改变当前 session 的 name
unset($_SESSION['name'])	删除名为 name 的 session
session_destroy()	结束当前会话，删除所有 session

注意　（1）如果要改变当前 session 的 name 值，必须在 session()之前调用 session_name()函数，而且 session_name 不能全部是数字，否则会不停地生成新的 session_id。

（2）不可以写成 unset($_SESSION)，这样会禁止整个会话的功能。

在本节最后，介绍使用 session 时要注意的几个问题。

1．尽可能地将 session_start()放到第 1 行

这种情况是新手最容易犯的错误。产生的错误代码为：

```
Warning: session_start() [function.session-start]: Cannot send session cache limiter - headers already sent…
```

其原因就是在使用 session_start()之前，就有 HTML 代码输出了。也许有的读者会说："没有，session_start()之前没有任何代码，绝对没有。"那么，请检查你的程序是不是有空行，或类似 echo 语句的输出。如果有，请去掉，因为就算是一个小小的空格都是不可以的。所以，为了避免这类错误的发生，尽可能地将 session_start()放到第 1 行。

2．在使用 session 之前一定要先写 session_start()

大多数读者在使用 session 之前都能先调用 session_start()函数，但对于 session_destroy()函数却经常忽略。session_destroy()虽然是结束当前会话并删除所有 session，但在删除之前，也要先开启 session 支持才可以，不然会产生这样的错误代码：

```
session_destroy() [function.session-destroy]: Trying to destroy uninitialized session in…
```

所以，凡是在使用 session 或 session 函数的页面中，都要加上 session_start()这句话。

3．删除所有 session

如果想删除所有 session，但又不想结束当前会话，用 unset 一个一个删除实在是太麻烦了，最简单的办法就是将一个空数组赋给$_SESSION，如$_SESSION = array()，这样就解决了。

7.6.3　添加文章的实现过程

添加文章模块使用的数据表：tb_article

当用户登录后，系统会直接进入到文章添加页（file.php），也可以通过单击"文章管理"/"添加博客文章"回到 file.php 页。添加文章页面的运行结果如图 7.21 所示。







图 7.21　添加文章页面的运行结果

添加文章页为一个发布表单，包括文章主题、文字编辑、文章内容等元素。部分表单元素如表 7.5 所示。

表 7.5　添加文章页面的主要表单元素

名　称	元素类型	重要属性	含　义
myform	form	method="post" action="check_file.php"	添加文章表单
txt_title	text	id="txt_title" size="68"	文章标题
font	select	name="font" class="wenbenkuang" id="font" onChange="showfont(this.options[this.selectedIndex].value)"	文章字体
size	select	class="wenbenkuang" onChange="showsize(this.options[this.selectedIndex].value)"	字体大小
color	select	onChange="showcolor(this.options[this.selectedIndex].value)" name="color" size="1" class="wenbenkuang" id="select"	字体颜色
file	textarea	cols="75" rows="20" id="file" style="border:0px;width:520px;"	文章内容
btn_tj	submit	id="btn_tj" value="提交" onClick="return check();"	“提交”按钮

当用户填写完博客主题和文章内容后，单击“提交”按钮，系统将跳转到处理页（check_file.php）进行处理。在处理页中，将传过来的文章标题、文章作者和文章内容等参数组成 insert 语句，并最终保存到数据表中。如果添加信息成功，系统返回到本页，可继续执行添加操作；如果添加失败，则返回到上一步。程序的关键代码如下：

例程02　代码位置：光盘\TM\07\tmlog\check_file.php

```
<?php
session_start();                              //开启 session 支持
include "Conn/conn.php";                      //包含数据库链接文件
if($btn_tj<>""){                              //判断传值页面
$title=$_POST[txt_title];                     //取得文章标题
$author=$_SESSION[username];                  //取得作者
$content=$_POST[file];                        //取得文章内容
$now=date("Y-m-d H:i:s");                     //使用 date()函数生成发布时间
/*生成 insert 语句*/
$sql="Insert Into tb_article (title,content,author,now) Values ('$title','$content','$author','$now')";
$result=mysql_query($sql);                    //执行 insert 语句
/*根据$result，返回结果*/
if($result){
     echo "<script>alert('恭喜您，你的文章发表成功!!!');window.location.href='file.php';</script>";
```

```
}
else{
      echo "<script>alert('对不起，添加操作失败!!!');history.go(-1);</script>";
}
}else{
      echo "<script>alert('对不起，添加操作失败!!!');history.go(-1);</script>";
}
?>
```

说明 文章添加页面中，使用了部分 UBB 语法，由于 UBB 语法不属于本书的范畴，所以这里不做讲解，请感兴趣的朋友查看相关的书籍。

7.6.4　文章列表的实现过程

查看文章列表使用的数据表：tb_article

单击“文章管理”/“我的文章”，将显示用户发表过的文章列表。文章列表页面（myfiles.php）的运行结果如图 7.22 所示。



图 7.22　文章列表页面的运行结果

文章列表页面使用了分页技术和 do…while 循环语句来输出文章标题。程序关键代码如下：

例程03　代码位置：光盘\TM\07\tmlog\myfiles.php

```
<?php
session_start();                                            //开启 session 支持
include "Conn/conn.php";                                    //包含数据库链接文件
include "check_login.php";                                  //包含权限检查文件
?>
/*分页*/
<?php
/*$page 为当前页，如果$page 为空，则初始化为 1*/
if ($page==""){
      $page=1;}
❶     if (is_numeric($page)){
      $page_size=20;                                        //每页显示 20 条记录
      $query="select count(*) as total from tb_article where author = '".$_SESSION[username]."' order by id
desc";
      $result=mysql_query($query);                          //查询符合条件的记录总条数
      $message_count=mysql_result($result,0,"total");       //要显示的总记录数
      $page_count=ceil($message_count/$page_size);          //根据记录总数除以每页显示的记录数求出所分的页数
      $offset=($page-1)*$page_size;                         //计算下一页从第几条数据开始循环
```

```
        $sql=mysql_query("select id,title from tb_article where author = '".$_SESSION[username]."' order by id
desc limit $offset, $page_size");
        $info=mysql_fetch_array($sql);
?>
…
<?php
/*输出结果集*/
if($info){
        $i=1;                                        //文章序号
❷       do{                                          //do…while 循环开始
?>
<tr>
<!--显示文章序号和文章标题-->
<td width="498" align="left" valign="top">    <a href="showmy.php?file_id=<?php echo
$info[id];?>"><?php echo $i."、".$info[title];?></a> </td>
</tr>
<?php
$i=$i+1;
}while($info=mysql_fetch_array($sql))                //循环结束
?>
</table></td>
</tr>
</table></td>
</tr>
<?php } ?>
…
<!--翻页条-->
<td width="33%">  页次：<?php echo $page;?>/<?php echo $page_count;?>页 记录：<?php
echo $message_count;?> 条  </td>
<td width="67%" align="right" class="hongse01">
<?php
/*如果当前页不是首页*/
if($page!=1){
/*显示"首页"超链接*/
echo  "<a href=myfiles.php?page=1>首页</a> ";
/*显示"上一页"超链接*/
echo "<a href=myfiles.php?page=".($page-1).">上一页</a> ";
}
/*如果当前页不是尾页*/
if($page<$page_count)
{
/*显示"下一页"超链接*/
echo "<a href=myfiles.php?page=".($page+1).">下一页</a> ";
/*显示"尾页"超链接*/
echo  "<a href=myfiles.php?page=".$page_count.">尾页</a>";
}}
?>
```

代码贴士

❶ is_numeric($page)：is_numeric()函数判断变量$page 是否为数字，如果是数字，则返回 true，否则返回 false。如果一组字符全部由数字组成，那么 is_numeric()函数也将返回 true。

❷ do…while 循环和 while 循环：do…while 循环是先执行{}中的代码段，然后判断 while 中的条件表达式是否成立，如果成立返回 true，重复输出{}中的内容，否则结束循环，执行 while 下面的语句。while 循环是先判断 while 中的表达式，当返回 true 时，再执行{}中的代码。两者的区别是：do…while 循环比 while 循环多输出 1 次结果。

7.6.5 查看文章、评论的实现过程

查看文章、评论使用的数据表：tb_article、tb_filecomment

单击列表中任意一个文章标题，都会看到对应的文章内容和文章评论。查看文章页面（showmy.php）的运行结果如图 7.23 所示。

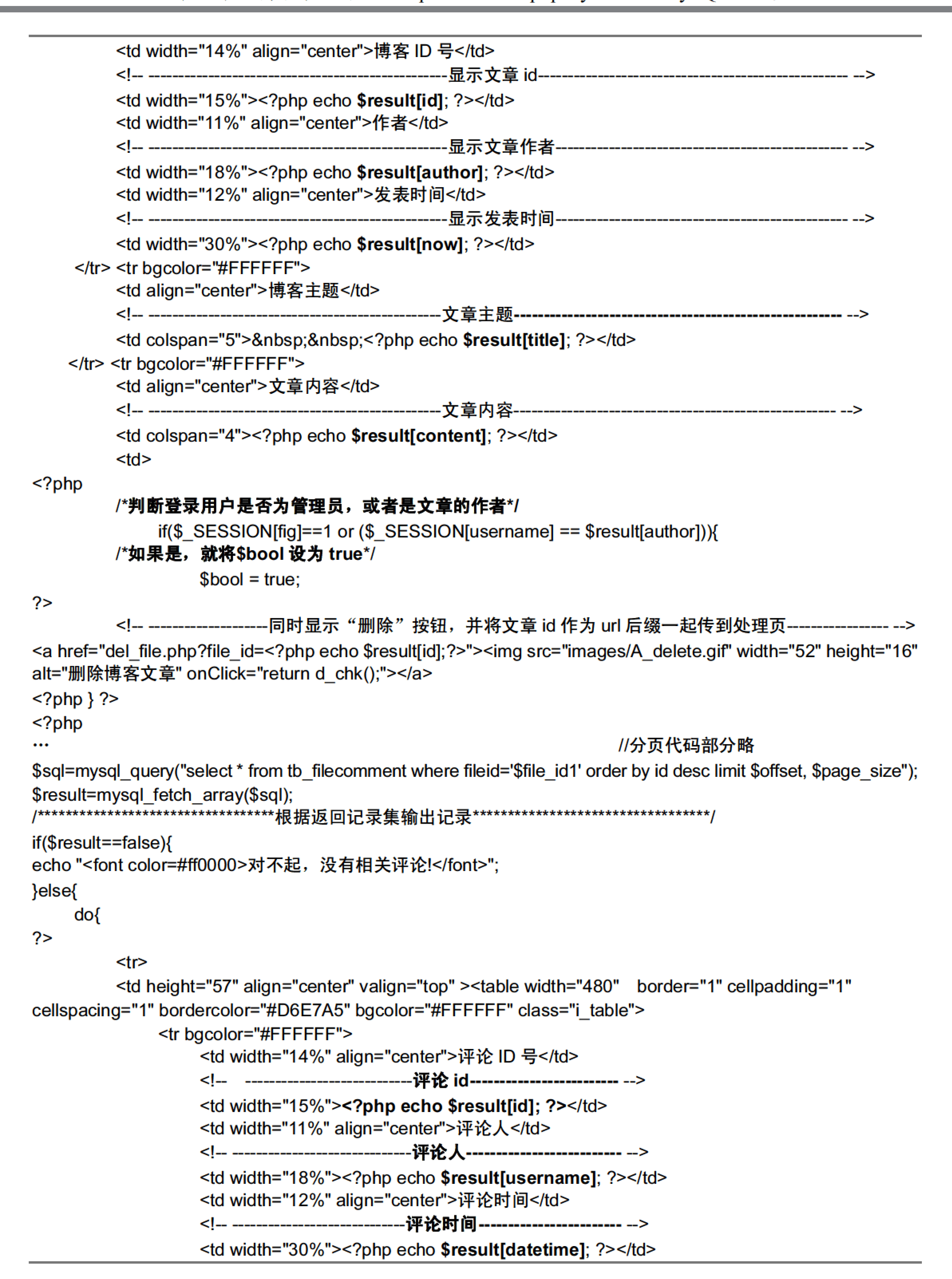


图 7.23 查看文章页面的运行结果

系统根据当前页面传过来的文章 id 值从数据表 tb_article 中返回对应的文章信息（包括文章 id、文章作者、文章标题、文章内容和发表时间）、输出文章信息后，开始查找表 tb_filecomment 中 fileid 字段值等于文章 id 的所有评论集，并通过分页显示出来。显示文章页面（showmy.php）的关键代码如下：

例程04 代码位置：光盘\TM\07\tmlog\showmy.php

```
<?php
session_start();                                          //开启 session 支持
include "Conn/conn.php";                                  //包含数据库链接文件
include "check_login.php";                                //包含权限检查文件
$file_id1=$_GET[file_id];                                 //取得文章 ID 号
$bool = false;                                            //判断用户删除权限，默认的 false 没有权限
?>
            <!--首部导航栏-->
…
<! -- ------------------------------------------ -- >
            <!--文章显示区-->
<?php
      $sql=mysql_query("select * from tb_article where id = ".$file_id1);  //根据文章 ID 号查找文章
      $result=mysql_fetch_array($sql);                    //返回结果集
?>
<!-- -----------------------------------------------------------显示文章----------------------------------------------------------- -->
<table width="100%"   border="1" cellpadding="1" cellspacing="1" bordercolor="#D6E7A5" bgcolor="#FFFFFF"
class="i_table">
      <tr bgcolor="#FFFFFF">
```



```
            <td width="14%" align="center">博客 ID 号</td>
            <!-- -------------------------------------------显示文章 id--------------------------------------------- -->
            <td width="15%"><?php echo $result[id]; ?></td>
            <td width="11%" align="center">作者</td>
            <!-- -------------------------------------------显示文章作者------------------------------------------ -->
            <td width="18%"><?php echo $result[author]; ?></td>
            <td width="12%" align="center">发表时间</td>
            <!-- -------------------------------------------显示发表时间------------------------------------------ -->
            <td width="30%"><?php echo $result[now]; ?></td>
      </tr> <tr bgcolor="#FFFFFF">
            <td align="center">博客主题</td>
            <!-- -------------------------------------------文章主题----------------------------------------------- -->
            <td colspan="5">  <?php echo $result[title]; ?></td>
     </tr> <tr bgcolor="#FFFFFF">
            <td align="center">文章内容</td>
            <!-- -------------------------------------------文章内容---------------------------------------------- -->
            <td colspan="4"><?php echo $result[content]; ?></td>
            <td>
<?php
            /*判断登录用户是否为管理员，或者是文章的作者*/
                  if($_SESSION[fig]==1 or ($_SESSION[username] == $result[author])){
            /*如果是，就将$bool 设为 true*/
                        $bool = true;
?>
            <!-- -----------------同时显示“删除”按钮，并将文章 id 作为 url 后缀一起传到处理页--------------- -->
<a href="del_file.php?file_id=<?php echo $result[id];?>"><img src="images/A_delete.gif" width="52" height="16"
alt="删除博客文章" onClick="return d_chk();"></a>
<?php } ?>
<?php
…                                                                                    //分页代码部分略
$sql=mysql_query("select * from tb_filecomment where fileid='$file_id1' order by id desc limit $offset, $page_size");
$result=mysql_fetch_array($sql);
/**********************************根据返回记录集输出记录**********************************/
if($result==false){
echo "<font color=#ff0000>对不起，没有相关评论!</font>";
}else{
      do{
?>
            <tr>
            <td height="57" align="center" valign="top" ><table width="480"   border="1" cellpadding="1"
cellspacing="1" bordercolor="#D6E7A5" bgcolor="#FFFFFF" class="i_table">
                  <tr bgcolor="#FFFFFF">
                        <td width="14%" align="center">评论 ID 号</td>
                        <!--   --------------------------评论 id------------------------ -->
                        <td width="15%"><?php echo $result[id]; ?></td>
                        <td width="11%" align="center">评论人</td>
                        <!-- --------------------------评论人------------------------ -->
                        <td width="18%"><?php echo $result[username]; ?></td>
                        <td width="12%" align="center">评论时间</td>
                        <!-- -------------------------评论时间------------------------ -->
                        <td width="30%"><?php echo $result[datetime]; ?></td>
```

```
                </tr>
                <tr bgcolor="#FFFFFF">
                    <td align="center">评论内容</td>
                    <!-- ---------------------------评论内容----------------------- -->
                    <td colspan="4"><?php echo $result[content]; ?></td>
                    <td>
<?php
            /*如果$bool 为真，输出删除超链接，否则跳过*/
if ($bool){
?>
            <a href="del_comment.php?comment_id=<?php echo $result[id]?>"><img src="images/A_delete.gif"
width="52" height="16" alt="删除博客文章评论" onClick="return d_chk();"></a>
<?php }?>
</td></tr> </table></td> </tr>
<?php
            }while($result=mysql_fetch_array($sql));
}
?>
</table>
…                                          <!--翻页功能代码部分略-->
```

7.6.6 删除文章、评论的实现过程

查看文章、评论模块使用的数据表：tb_article、tb_filecomment

在查看文章评论页面，当系统判定当前用户为管理员或文章作者时，在每篇文章和评论的后面，都将显示相应的“删除”按钮。单击任意的“删除”按钮，系统会提示是否删除，如果确认，将跳转到处理页（del_file.php 和 del_comment.php），完成删除操作。

在删除文章的处理页中，删除文章的同时，也删除了该篇文章相关的评论。处理页首先在文章列表（tb_article）中删除 id 等于$file_id 的记录，如果没有可删除记录，则提示失败，并返回上一步；如果删除成功，则转到评论列表（tb_filecomment）中，删除所有该篇文章的评论。删除文章页（del_file.php）的关键代码如下：

例程05　代码位置：光盘\TM\07\tmlog\del_file.php

```
<?php
    session_start();                                                    //开启 session 支持
    include "check_login.php";                                          //包含权限检查文件
    include "Conn/conn.php";                                            //包含数据库链接文件
   $sql="delete from tb_article where id=".$file_id;                    //删除文章的 delete 语句
   $result=mysql_query($sql,$link);                                     //执行删除操作
    if($result){
        $sql1 = "delete from tb_filecomment where fileid = ".$file_id;  //删除相对应的文章评论
        $rst1 = mysql_query($sql1,$link);                               //执行删除评论的操作
        if($rst1)
            echo "<script>alert('博客文章已被删除!');location='myfiles.php]';</script>";
        else
            echo "<script>alert('删除失败!');history.go(-1);</script>";
    }
```

```
	else{
		echo "<script>alert('博客文章删除操作失败!');history.go(-1);</script>";
	}
?>
```

因为删除文章的过程也包含了删除评论的过程，所以这里就不给出删除评论的代码了。

7.7　图片上传模块设计

由于动态网络编程技术的诞生，使得网络更加人性化，为了能够和用户更好地互动，很多网站都提供了让用户上传图片的功能。一个网站拥有图片上传功能是非常必要的。

7.7.1　图片上传模块概述

图片上传在动态网页开发过程中应用非常广泛。如果有比较好的图片想和其他人一同分享，就可以通过图片上传功能来实现，以增加网站的核心竞争力。本系统的图片上传模块主要实现对图片的添加、浏览、查询和删除操作，而对图片的删除则只有管理员才有权限。图片上传模块框架图如图 7.24 所示。

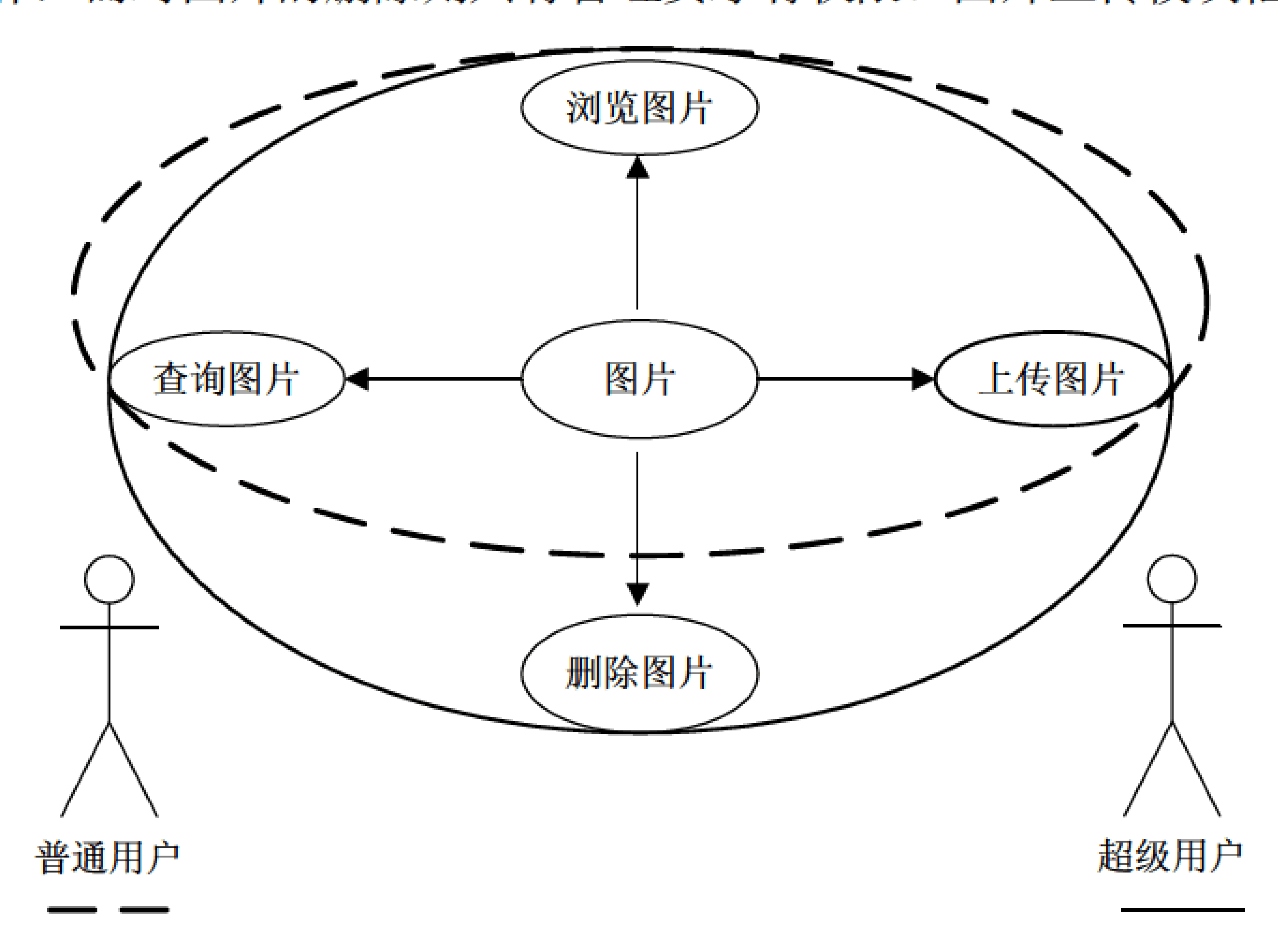


图 7.24　图片上传模块框架图

7.7.2　图片上传模块技术分析

既然是图片上传模块，显而易见本节的主要技术就是上传图片功能了。上传图片和上传文件的原理基本相同，下面就来学习如何上传图片和图片的两种保存方式。

1．上传图片的基本流程

在网页中实现上传图片功能的步骤如下：

（1）通过<form>表单中的 file 元素选取上传数据。

使用 file 元素上传数据时注意一点：就是在 form 表单中要加上属性 enctype="multipart/form-data"，

否则上传不了文件（图片）。

（2）在处理页中使用$_FILES 变量中的属性判断上传文件类型和上传文件（图片）大小是否符合要求。

$_FILES 变量为系统预定义变量，保存的是上传文件（图片）的相关属性。使用格式如下：

```
$_FILES[name][property];
```

$_FILES 的相关属性如表 7.6 所示。

表 7.6 $_FILES 的相关属性

属 性 值	说 明
name	上传文件的文件名
type	上传文件的类型
size	上传文件的大小
tmp_name	上传文件在服务器中的临时文件名
error	上传文件失败的错误代码

（3）使用 move_uploaded_file()函数上传文件（图片）或将文件（图片）以二进制的形式保存到数据库中。

使用函数将文件（图片）保存到对应的文件夹中和以二进制的形式保存到数据库中是上传文件（图片）的两种形式，稍后将单独做介绍。

（4）返回页面等待下一步操作。

2．使用上传函数保存文件（图片）

使用上传函数上传文件（图片）的本质就是将文件（图片）从浏览器端复制到服务器端指定的文件夹中，数据库所存储的就是文件（图片）的相对地址。当页面显示图片时，实际是分两步：第一步是读取数据表中的地址；第二步是根据地址找到并在页面中显示图片。使用目录保存文件的好处是减少了数据库的容量和对数据库的压力，而且图片很容易被搜索引擎抓到，从而提高网站的流量和人气。

move_uploaded_file()函数的一般格式如下：

```
bool move_uploaded_file ( string filename, string destination );
```

参数说明如下。

☑ filename：上传到服务器中的临时文件名。

☑ destination：保存文件的实际路径。

注意 这里的 filename 为临时文件名，而不是上传文件的原文件名，可以通过$_FILES[filename][tmp_name]来获取。

下面看一个实例，程序代码如下：

```
<label>请选择要上传的图片（文件小于 500K）：</label>
<!--上传文件的 form 表单-->
```



```
<form method="post" action="#" enctype="multipart/form-data">
    <input type="hidden" name="action" value="upload" />
    <input type="file" name="u_file"/>
    <input type="submit" value="上传" />
</form>
<!------------------------------------>
<?php
    /*判断是否为上传动作*/
    if(isset($_POST[action])){
        $file_path = "./";                                           //上传文件存放路径，./为当前目录下
        $profix = array(".jpg",".gif",".jpeg");                      //设置允许上传的文件后缀类型
        $f_name = $_FILES[u_file][name];                             //取得要上传的文件名
        $pro_name=substr($f_name,strrpos($f_name,"."));              //取得上传文件的后缀
        if(!in_array($pro_name, $profix)){                           //判断上传文件的类型是否为允许类型
            echo "文件格式不对";
            exit();
        }
        if($_FILES[u_file][size] <= 0){                              //判断上传文件的大小
            echo "文件上传错误，或文件大于 500KB,请重新上传";         //如果文件过大，提示错误
        }
        else{
            /*上传文件函数*/
            move_uploaded_file($_FILES[u_file][tmp_name],$file_path.$_FILES[u_file][name]);
            echo "上传成功!";
        }
?>
```

3．使用二进制保存图片

上传图片的另一种保存方式是以二进制的形式存储在数据库中。在计算机看来，再美的图片、再感人的文章，也不过就是一堆“0111010011…”的代码段而已，和其他数据没有什么区别，存到数据表中都是一样的。存储图片不再需要使用 move_uploads_file()函数，在处理页中直接使用二进制的形式读取出文件图片，并存到数据表中。本系统采用的就是二进制的形式来保存图片，具体操作将在实现过程中给出。

使用二进制来保存图片的好处是安全，特别是涉及个人隐私，不易被窃取；存储方便，和表中其他数据同等操作就可以；节省磁盘空间，由于文件系统类型的限制，放到数据表中的数据一定比直接放到磁盘中要节省空间。

当然二进制形式的图片也有很大弊端，在传出的过程中，如果某个二进制位出现丢失或损坏，那么整张图片将无法显示，而是变为一堆乱码，而且如果存储的图片过多，造成数据库过于庞大，将影响其他数据的正常读取，增大服务器的负担。所以，对于太大的文件，就不适宜保存到数据库中，而是应该放到文件夹。

7.7.3　图片上传的实现过程

图片上传使用的数据表：tb_tpsc

博客用户登录后，单击导航栏中的“图片管理”/“添加图片”选项，即可进入添加图片页面，在“图片名称”文本框中添加上传的图片名称，在“上传路径”文本框中选择或者单击“浏览”按钮选

择自己喜欢的图片，单击“提交”按钮，以二进制形式将图片上传到数据库中。图片上传页面的运行结果如图 7.25 所示。



图 7.25　图片上传页面的运行结果

图片上传页是一个上传文件的表单，主要包括一个文本域、一个文件域和一个“提交”按钮。部分表单元素的名称及属性如表 7.7 所示。

表 7.7　图片上传页面中的表单元素

名　称	元素类型	重要属性	含　义
myform	form	method="post" action="tptj_ok.php"　enctype="multipart/form-data"	图片上传表单
tpmc	text	type="text" id="tpmc" size="40"	图片名称
file	file	type="file" size="23" maxlength="60"	上传路径
btn_tj	submit	type="submit" id="btn_tj" value="提交" onClick="return pic_chk();"	“提交”按钮

当用户输入图片名称，并选择图片路径后，单击“提交”按钮，系统将进入到上传处理页（tptj_ok.php）中进行处理。在处理页中，首先对图片名称进行处理，去掉特殊字符、空行和空格，然后对上传的文件进行类型检查和文件大小检查。最后，以二进制的形式和图片的其他信息（如上传用户、上传时间等）一起存进数据表中。关键代码如下：

例程06　代码位置：光盘\TM\07\tmlog\tptj_ok.php

```
<?php
session_start();                                            //开启 session 支持
include "check_login.php";                                  //包含权限检查文件
include "Conn/conn.php";                                    //包含数据库链接文件
if($_POST["btn_tj"]=="提交"){
❶           $tpmc=htmlspecialchars($tpmc);                  //将图片名称中的特殊字符转换成 HTML 格式
❷       $tpmc=str_replace("\n","<br>",$tpmc);               //将图片名称中的回车符以自动换行符取代
        $tpmc=str_replace(""," ",$tpmc);               //将图片名称中的空格以“ ”取代
         $author=$_SESSION[username];
        $scsj=date("Y-m-d");                                //设置图片的上传时间
         $profix = array(".jpg",".gif",".jpeg");            //设置允许上传的文件后缀类型
         $f_name = $_FILES[file][name];                     //取得要上传的文件名
         $pro_name=substr($f_name,strrpos($f_name,"."));    //取得上传文件的后缀
         /*判断上传文件的类型是否为允许类型*/
❸        if(!in_array(strtolower($pro_name), $profix)){
              echo "<script>alert('文件格式不对');history.go(-1);</script>";
              exit();
         }
         /*判断上传文件的大小，如果文件过大，提示错误*/
```



```
			if($_FILES[file][size] > 500000){
				echo "<script>alert('文件上传错误,请重新上传');history.go(-1)</script>";
				exit();
			}
			else{
			$fp=fopen($file,"r");                                    //以只读方式打开文件
			$file=addslashes(fread($fp,filesize($file)));            //将文件中的引号部分加上反斜线
			$query="insert into tb_tpsc (tpmc,file,author,scsj) values ('$tpmc','$file','$author','$scsj')";
			$result=mysql_query($query);
			echo "<meta http-equiv=\"refresh\" content=\"1;url=browse_pic.php\">图片上传成功，请稍等...";
		}
}
?>
```

代码贴士

❶ htmlspecialchars()函数：将特殊字符转换成 HTML 格式。

❷ str_replace()函数：取代所有在字串中出现的字串。语法如下：

```
mixed str_replace ( mixed search, mixed replace, mixed subject , int &count )
```

str_replace()函数将所有在参数 subject 中出现的 search 以参数 replace 替换，参数&count 表示替换字符串执行的次数。

❸ strtolower：将字符转换为小写字母。

7.7.4 图片浏览的实现过程

图片上传使用的数据表：tb_tpsc

无论是注册用户，还是非注册用户，只要登录网站，就可以无条件地浏览所有图片。而删除图片除了管理员，其他人都无权操作。非注册用户可以通过首页中的“最新图片”进入图片浏览页面，注册用户先进入个人管理界面，单击“图片管理”/“浏览图片”菜单，同样可以进入图片浏览页面。注册用户浏览图片页面的运行结果如图 7.26 所示。



图 7.26　浏览图片页面的运行结果

本页的实现代码和查看文章页面略有不同，查看文章页面中，每条数据占了一行，而查看图片则采用的是分栏显示，以每行两张图片的格式输出，每页显示 4 张图片。通过单击小图片，可以查看图片原效果图。程序关键代码如下：

例程07　代码位置：光盘\TM\07\tmlog\browse_pic.php

```
<?php
	session_start();                              //开启 session 支持
	include "Conn/conn.php";                      //包含数据库链接文件
	include "check_login.php";                    //包含权限检查文件
?>
…                                                 <!--首部导航栏-->
```

```
<tr align="left" colspan="2" >
    <td width="390" height="25" colspan="3" valign="top" bgcolor="#EFF7DE"> <span class="tableBorder_
LTR">浏览图片</span></td></tr>
    <tr>
    <td height="192" align="center" valign="top" >
    <?php
    …                                                   //分页显示
        $query="select * from tb_tpsc where scsj order by id desc limit $offset, $page_size";
        /*返回结果集*/
        $result=mysql_query($query);
    ?>
    <table width="496" border="1" align="center" cellpadding="3" cellspacing="1" bordercolor="#D6D7D6">
      <tr>
       <?php
        /*设置一个变量，判断当前数据是奇数输出还是偶数输出*/
            $i=1;
          while($info=mysql_fetch_array($result))
          {
        /*如果当前为偶数输出，记录输出完毕后结束<tr>标签*/
            if($i%2==0){
            ?>
          <td width="500"><table width="245" border="0" cellpadding="0" cellspacing="0">
              <tr>
                <td colspan="2"><div align="center">
                    <!--显示图片页 image.php -->
                      <a href="image.php?recid=<?php echo $info[id]; ?>" target="_blank"><img src=
"image.php?recid=<?php echo $info[id];?>" width="150" height="100"></a></div></td></tr>
                    <tr>
                      <td width="109" height="25" align="left"> 图片名称:<?php echo $info[tpmc];?>
</td>
                      <td width="128">上传时间:<?php echo $info[scsj];?></td></tr>
                        <tr>
                          <td colspan="2" height="25">
                          <?php
                          /*如果登录用户为管理员，显示删除操作*/
                                if ($_SESSION[fig]==1){
                          ?>
                            <a href="remove.php?pic_id=<?php echo $info[id]?>"><img src="images/
A_delete.gif" width="52" height="16" alt="删除图片" onClick="return pic_chk();"></a>
                          <?php
                                }
                           ?>
                          </td>
                        </tr>
                </table></td>
              </tr>
              <?php
          }
        /*否则，开始新的一行*/
            else
            {
```

```
                    ?>
  <td width="500" ><table width="236" border="0" cellpadding="0" cellspacing="0">
          <tr>
            <td colspan="2"><div align="center"><a href="image.php?recid=<?php echo $info[id]; ?>" target="_blank">
<img src="image.php?recid=<?php echo $info[id];?>" width="150" height="100"></a></div></td>
          </tr>
          <tr>
            <td width="110" height="25"> 图片名称:<?php echo $info[tpmc];?></td>
            <td width="126">上传时间: <?php echo $info[scsj];?></td>
          </tr>
            <tr>
                  <td colspan="2" height="25">
                  <?php
                                    if ($_SESSION[fig]==1){
                                ?>
                                  <a href="remove.php?pic_id=<?php echo $info[id]?>"><img src="images/
A_delete.gif" width="52" height="16" alt="删除图片" onClick="return pic_chk();"></a>
                                <?php
                                    }
                                 ?>
                  </td>
            </tr>
      </table></td>
        <?php
                }
                  /*变量$i 自加 1，while 循环结束*/
                   $i++;
                  }
                ?>
  </tr>
```

通过上述代码可以看到，预览图片和显示图片实际调用的都是同一页面：image.php。image.php 页就是用来显示图片的实际效果的。实现代码如下：

例程08　代码位置：光盘\TM\07\tmlog\image.php

```
<?php
    include "Conn/conn.php";                                                  //包含数据库链接文件
   $query="select id,tpmc,file,author,scsj from tb_tpsc where id=".$recid;    //根据 id 生成查询语句
   $result=mysql_query($query);                                               //执行查询语句
   if(!$result) die("error: mysql query");                                    //判断是否有返回结果
   $num=mysql_num_rows($result);                                              //计算结果集个数
   if($num<1) die("error: no this recorder");
   $data = mysql_result($result,0,"file");                                    //返回图片数据
    echo $data;                                                               //输出图片
?>
```



mysql_result()函数不能和其他处理结果集函数混合调用。

7.7.5 删除图片的实现过程

删除图片使用的数据表：tb_tpsc

删除图片是管理员才有的权限，在图片浏览的实现代码中，已经给出如何判断登录用户是否为管理员。当管理员单击“删除”超链接时，处理页（remove.php）会根据传过来的 id，删除对应的数据表中的数据。实现代码如下：

例程09　代码位置：光盘\TM\07\tmlog\remove.php

```
<?php
	session_start();                                                        //开启 session 支持
	include "check_login.php";                                              //包含权限检查文件
	include "Conn/conn.php";                                                //包含数据库链接文件
	if($_session[fig] != "1"){                                              //判断是否为管理员
		echo "<script>alert('您没有删除权限 ');history.go(-1);</script>";
	}
	$sql="delete from tb_tpsc where id=".$pic_id;                           //生成删除语句
	$result=mysql_query($sql);                                              //执行删除语句
	if($result){                                                            //根据$result 返回结果
		echo "<script>alert('图片删除成功!');location='browse_pic.php';</script>";
	}
	else{
		echo "<script>alert('图片删除操作失败!');history.go(-1);</script>";
	}
?>
```

7.7.6 单元测试

在进行上传模块的测试时发现：当上传比较大的图片时，文件大小的验证部分并没有生效，而是直接跳过。在处理页中产生错误，并且显示页面不显示图片。运行结果如图 7.27 所示。

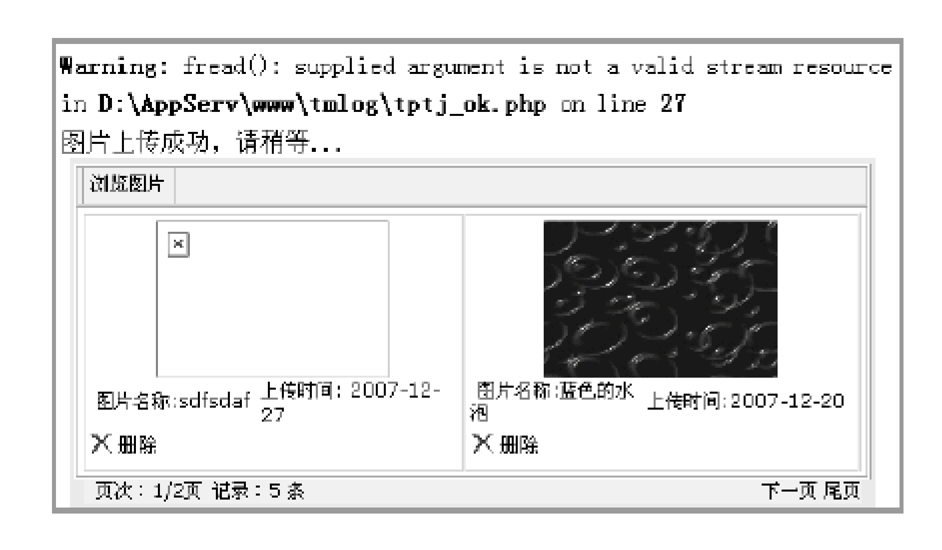


图 7.27　上传图片错误信息及显示结果

查看原文件大小验证的代码如下：

```
/*判断上传文件的大小，如果文件过大，提示错误*/
if($_FILES[file][size] > 500000){
	echo "<script>alert('文件上传错误，请重新上传');history.go(-1)</script>";
	exit();
}
```



经反复测试后发现：如果上传图片大于 500KB 而小于 2MB 时，原代码生效。如果图片大于 2MB，则代码被跳过，这时输出 $_FILES[file][size] 的值为 0。原来，在配置文件 php.ini 中，upload_mix_filesize=2MB，也就是说，上传文件的大小超过了系统设置的值，那么$_FILES[file][size] 返回 0，在判断表达式中返回 true，代码自然被跳过。

改写验证代码，更改后的代码如下：

例程10　代码位置：光盘\TM\07\tmlog\tptj_ok.php

```
if(($_FILES[file][size] <= 0) or ($_FILES[file][size] > 500000)){
		echo "<script>alert('文件上传错误，或文件大于 500KB，请重新上传');history.go(-1)</script>";
		exit();
}
```

进行二次测试，问题没有再出现，系统正常运行。

7.8　朋友圈模块设计

7.8.1　朋友圈模块概述

本系统的朋友圈模块的主要功能是添加、查询、删除好友，添加的好友除了该用户以外，包括管理员在内的所有外人都不可以查看，以保证其个人隐私不被外泄。用户被删除时，该用户现有的朋友圈也一并被删除。朋友圈模块框架图如图 7.28 所示。

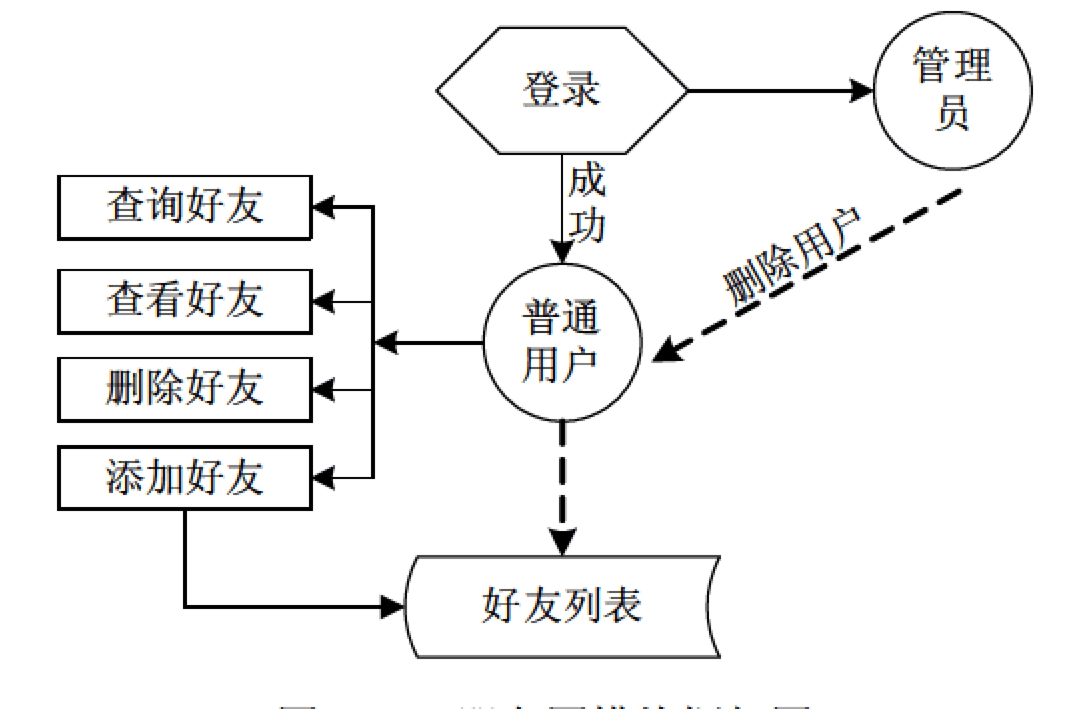


图 7.28　朋友圈模块框架图

7.8.2　朋友圈模块技术分析

在查询好友的功能中，使用到了模糊查询语句，用于模糊查找好友列表。模糊查询语句使用的是 like 运算符。在 PHP 中，带有 like 运算符的查询语句的常用格式有以下两种：

（1）使用通配符“%”的 where 子句

通配符“%”表示 0 个或多个任意长度和类型的字符，包括中文汉字。

示例 1：表示查找所有内容包含“好”字的文章。

```
select * from tb_file where content like '%好%';
```

示例 2：查找所有包含“好”字或“高”字的文章，这时可以配合 or 运算符来使用。代码如下：

```
select * from tb_file where content like '%好%' or content like '%高%';
```

（2）使用通配符“_”的 where 子句

通配符“_”表示匹配任意的单个字符。

示例 1：查找用户名只包含 5 个字符，其中后 4 个字符为 soft 的用户。代码如下：

```
select * from tb_user where regname like '_soft';
```

示例 2：查找所有以 t 开头，并且以 t 结尾的、中间包含 3 个字符的用户。代码如下：

```
select * from tb_user where regname like 't___t';
```

查找的结果为：tsoft。

注意 使用 MySQL 做模糊查询要注意编码问题。如果编码不统一，那么查询时就容易查不到数据，或返回的数据不匹配。所以在安装 MySQL 时，要保持和系统编码的统一。常用的编码格式有 gb2312、ISO-8859-1、utf8 和 gbk 等。

7.8.3　查询好友的实现过程

查询好友使用的数据表：tb_friend

当用户要查询好友时，单击“朋友圈管理”/“查询朋友信息”，显示查询页面。查询可以分为姓名查询和编号查询，均为模糊查询。当用户输入要查找的关键字后，单击“检索”按钮，或按 Enter 键，系统跳到处理页进行处理。查询好友页面的运行结果如图 7.29 所示。



图 7.29　查询好友页面的运行结果

查询页包含一个查询表单，包括查询条件和查询关键字两部分表单元素。主要表单元素如表 7.8 所示。

表 7.8　查询页表单的主要元素属性

名　称	元 素 类 型	重 要 属 性	含　义
myform	form	method="post" action="query_friend.php" onSubmit="return check();"	查询好友表单
sel_tj	select	<option value="name" selected>姓名</option> <option value="id">编号</option>	查询条件选择
sel_key	text	id="sel_key" size="30"	查询关键字
submit	submit	type="submit" name="submit" value="检索"	“检索”按钮

当处理页接收到查询条件及查询关键字后，生成模糊查询语句，执行 SQL 语句并返回查询结果。如果没有输入关键字，则弹出提示框；如果没有查找到任何结果，则输出“Sorry!没有您要找的朋友!”。

处理页的关键代码如下：

例程11　代码位置：光盘\TM\07\tmlog\query_friend.php

```
<?php
session_start();                                    //开启 session 支持
include "Conn/conn.php";                            //包含数据库链接文件
include "check_login.php";                          //包含权限检查文件
?>
<script language="javascript">
/*JavaScript 脚本，验证输入信息，如果为空，则弹出提示框*/
function check(form){
        if (document.myform.sel_key.value==""){
                alert("请输入查询条件!");
                myform.sel_key.focus();
                return false;
        }
}
</script>
…
<?php
        if ($_POST[sel_key]!=""){
                $tj=$_POST[sel_tj];                 //接收查询字段
                $key=$_POST[sel_key];               //接收查询关键字
                $sql=mysql_query("select * from tb_friend where $tj like '%$key%' and username='$_SESSION
[username]'");
                $result=mysql_fetch_array($sql);    //执行查询语句
                if($result==false){
                        echo ("[<font color=red>Sorry!没有您要找的朋友!</font>]");
                }else{
?>
…                                                   <!--显示记录表单->
<?php }?>
```

7.9　开发技巧与难点分析

想象一下：动态的下拉菜单，半透明的背景效果，鼠标划过时的色彩交替，给人一种亦真亦幻的视觉冲击和美的感受，这样的网站是不是具有更大的吸引力？本系统个人管理页面中的管理菜单实现的就是这种效果。当用户登录到博客网站后，将鼠标移动到“朋友圈管理”等导航链接上时，将在下方显示出半透明的下拉式菜单，透过此下拉菜单仍可以看到页面上的内容，运行效果如图 7.30 所示。



图 7.30　动态半透明效果的下拉菜单

实现半透明背景的下拉菜单，首先需要在页面中实现下拉菜单，然后再通过设置下拉菜单的 CSS 样式实现半透明效果。实现下拉菜单的半透明效果可以应用 CSS 样式的透明效果滤镜 alpha 实现。alpha 属性是把一个目标元素与背景混合。这种“与背景混合”，通俗地说，是一个元素的透明度。透明效果滤镜 alpha 的语法如下：

```
{filter:alpha(opacity=opacity,finishopacity=finishopacity,style=style,startx=startx,starty=starty,finishx=finishx,finishy
=finishy)}
```

滤镜 alpha 的各个参数说明如表 7.9 所示。

表 7.9　alpha 滤镜的各个参数及说明

属　　性	说　　明
opacity	代表透明度水准。默认的范围是 0～100，其实是百分比的形式，也就是 0 代表完全透明，100 代表完全不透明
finishopacity	可选，如果想要设置渐变的透明效果，可以使用该参数指定结束时的透明度。范围也是 0～100
style	指定透明区域的形状特征，其中 0 代表统一形状、1 代表线形、2 代表放射状、3 代表长方形
startx	代表渐变透明效果的开始 X 坐标
starty	代表渐变透明效果的开始 Y 坐标
finishx	代表渐变透明效果的结束 X 坐标
finishy	代表渐变透明效果的结束 Y 坐标

在本实例中的 menuskin 类中，设置了 alpha 滤镜的属性。代码如下：

例程12　代码位置：光盘\TM\07\tmlog\css\style.css

```
.menuskin {
        BORDER: #666666 1px solid;                         //层边框样式
        VISIBILITY: hidden;                                //层可见为隐藏
        FONT: 12px Verdana;                                //层中的字体样式和大小
        POSITION: absolute;                                //定位方式
        background-image:url("../images/item_out.gif");    //背景图片
        background-repeat : repeat-y;                      //图片是否可重复
        filter: alpha(Opacity=85);                         //设置 alpha 的 opacity 属性等于 85，即透明度为 85%
}
```

在显示页面中显示菜单项的表单引入这个 menuskin 类，当鼠标划过特定的文字时，将会发现自动下拉菜单的效果为半透明的样式。页面关键代码如下：

例程13　代码位置：光盘\TM\07\tmlog\file.php

```
<!--弹出层设置-->
<div class=menuskin id=popmenu
        <!--鼠标进入该区域时调用的 js 方法-->
        onmouseover="clearhidemenu();highlightmenu(event,'on')"
        <!--鼠标离开该区域时调用的 js 方法-->
        onmouseout="highlightmenu(event,'off');dynamichide(event)"
        <!--设置层叠顺序和定位方式-->
        style="Z-index:100;position:absolute;">
</div>
```



```
…
<!--半透明的动态下拉菜单-->
<TABLE style="WIDTH: 580px" VERTICAL-ALIGN: text-top; cellSpacing=0 cellPadding=0 align="center">
	<TR align="center" valign="middle">
		<!--显示登录用户-->
		<TD style="WIDTH: 100px; COLOR: red;">欢迎您: <?php echo
$_SESSION[username]; ?>  </TD>
		<!--显示首页超链接-->
		<TD style="WIDTH: 80px; COLOR: red;"><SPAN   style="FONT-SIZE: 9pt; COLOR: #cc0033">
</SPAN><a href="index.php">博客首页</a></TD>
		<!--显示文章管理下拉菜单-->
		<TD style="WIDTH: 80px; COLOR: red;"><a   onmouseover=showmenu(event,productmenu)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand" >文章管理</a></TD>
		<!--显示图片管理下拉菜单-->
<TD style="WIDTH: 80px; COLOR: red;"><a   onmouseover=showmenu(event,Honourmenu)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand">图片管理</a></TD>
		<!--显示朋友圈管理下拉菜单-->
		<TD style="WIDTH: 90px; COLOR: red;"><a   onmouseover=showmenu(event,myfriend)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand" >朋友圈管理</a></TD>
<?php
		/*如果登录用户为管理员*/
		if($_SESSION[fig]==1){
?>
		<!-- ------------------------显示管理员管理下拉菜单------------------------------- -->
		<TD style="WIDTH: 80px; COLOR: red;"><a onmouseover=showmenu(event,myuser)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand" >管理员管理</a></TD>
<?php
		}
?>
		<!-- --------------------------------显示退出超链接------------------------------ -->
		<TD style="WIDTH: 80px; COLOR: red;"><a href="safe.php">退出登录</a></TD>
	</TR>
</TABLE>
```

通过代码可以发现，在每个超链接标签<a>中，都有 3 个参数，分别介绍如下。

☑ onMouseOver：鼠标进入该区域时的效果。本页面调用了 js 方法 showmenu()。

☑ onMouseOut：鼠标移出该区域时的效果。本页面调用 js 方法 delayhidemenu()。

☑ style：标签样式，本页面样式为“CURSOR:hand;”，显示一个手的形状。

调用的 js 方法就是实现下拉菜单的 JavaScript 代码：显示和隐藏菜单。由于这段代码是通用型代码，所以单独放到一个 js 文件中，方便其他页调用，js 文件名为 menu.js。调用时在页面中加入如下代码即可。

```
<script src="JS/menu.js"></script>
```

menu.js 文件中显示、隐藏菜单和变换背景等关键函数的代码如下：

例程14 代码位置：光盘\TM\07\tmlog\js\menu.js

```
/*浏览器版本号检测*/
var IE4=document.all&&navigator.userAgent.indexOf("Opera")==-1
```

```
var netscape6=document.getElementById&&!document.all
var netscape4=document.layers
        /*显示下拉菜单函数*/
function showmenu(e,vmenu,mod){
        /*判断浏览器版本*/
        if (!document.all&&!document.getElementById&&!document.layers)
                return
        /*菜单内容*/
        which=vmenu
        /*调用 clearhidemenu()函数*/
        clearhidemenu()
        /*调用 IE_clearshadow()函数*/
        IE_clearshadow()
        /*根据不同的浏览器，声明弹出式菜单*/
        menuobj=IE4? document.all.popmenu : netscape6? document.getElementById("popmenu") : netscape4?
document.popmenu : ""
        menuobj.thestyle=(IE4||netscape6)? menuobj.style : menuobj
        if (IE4||netscape6)
        /*如果是 IE 4 或 Netscape 6 浏览器，就调用 innerHTML 方法输出下拉菜单*/
                menuobj.innerHTML=which
        else{
        /*否则使用 write 方法输出下拉菜单*/
                menuobj.document.write('<layer name="other" bgColor="#E6E6E6" width="165"
onmouseover="clearhidemenu()" onmouseout="hidemenu()">'+which+'</layer>')
        /*关闭数据流*/
                menuobj.document.close()
        }
        /*调用不同的方法属性，来设置不同浏览器的宽、高及内部表格的大小、位置*/
        menuobj.contentwidth=(IE4||netscape6)? menuobj.offsetWidth : menuobj.document.other.document.width
        menuobj.contentheight=(IE4||netscape6)? menuobj.offsetHeight : menuobj.document.other.document. height
        eventX=IE4? event.clientX : netscape6? e.clientX : e.x
        eventY=IE4? event.clientY : netscape6? e.clientY : e.y
        var rightedge=IE4? document.body.clientWidth-eventX : window.innerWidth-eventX
        var bottomedge=IE4? document.body.clientHeight-eventY : window.innerHeight-eventY
                if (rightedge<menuobj.contentwidth)
                        menuobj.thestyle.left=IE4? document.body.scrollLeft+eventX-menuobj.contentwidth+menuOffX :
netscape6? window.pageXOffset+eventX-menuobj.contentwidth : eventX-menuobj.contentwidth
                else
                        menuobj.thestyle.left=IE4? IE_x(event.srcElement)+menuOffX : netscape6? window.pageXOffset+eventX :
eventX
                if (bottomedge<menuobj.contentheight&&mod!=0)
                        menuobj.thestyle.top=IE4? document.body.scrollTop+eventY-menuobj.contentheight-event. offsetY+
menuOffY-23 : netscape6? window.pageYOffset+eventY-menuobj.contentheight-10 : eventY-menuobj.contentheight
                else
                        menuobj.thestyle.top=IE4? IE_y(event.srcElement)+menuOffY : netscape6? window.pageYOffset+
eventY+10 : eventY
/*设置标签样式为可见*/
        menuobj.thestyle.visibility="visible"
/*调用 IE_dropshadow()方法*/
        IE_dropshadow(menuobj,"#999999",3)
        return false
```

```
}
…
/*隐藏下拉菜单函数*/
function hidemenu(){
	if (window.menuobj)
		menuobj.thestyle.visibility=(IE4||netscape6)? "hidden" : "hide"
	IE_clearshadow()
}
function dynamichide(e){
	if (IE4&&!menuobj.contains(e.toElement))
		hidemenu()
	else if (netscape6&&e.currentTarget!= e.relatedTarget&& !contains_netscape6(e.currentTarget, e.relatedTarget))
		hidemenu()
}
/*延迟隐藏下拉菜单函数*/
function delayhidemenu(){
	if (IE4||netscape6||netscape4)
		delayhide=setTimeout("hidemenu()",500)
}
/*停止隐藏菜单函数*/
function clearhidemenu(){
	if (window.delayhide)
		clearTimeout(delayhide)
}
…
/*设置菜单背景*/
/*鼠标移入该区域时显示的背景*/
function overbg(tdbg){
tdbg.style.background='url( images/item_over.gif)'
tdbg.style.border=' #9CA6C6 1px solid'
}
/*鼠标移出该区域时显示的背景*/
function outbg(tdbg){
tdbg.style.background='url( images/item_out.gif)'
tdbg.style.border=''
}
/*下拉菜单内容*/
/*文章管理下拉菜单*/
var productmenu='<table width=90><tr><td id=fileadd onMouseOver=overbg(fileadd) onMouseOut=outbg (fileadd)><a href=file.php>添加博客文章</a></td></tr>\
<tr><td id=query onMouseOver=overbg(query) onMouseOut=outbg(query)><a href=query.php>查询博客文章</a></td></tr>\
<tr><td id=myfiles onMouseOver=overbg(myfiles) onMouseOut=outbg(myfiles)><a href=myfiles.php>我的文章</a></td></tr></table>'
/*图片管理下拉菜单*/
var Honourmenu='<table width=90><tr><td id=picadd onMouseOver=overbg(picadd) onMouseOut= outbg (picadd)><a href=add_pic.php>添加图片</a></td></tr>\
<tr><td id=browse onMouseOver=overbg(browse) onMouseOut=outbg(browse)><a href=browse_pic.php>浏览图片</a></td></tr>\
<tr><td id=querypic onMouseOver=overbg(querypic) onMouseOut=outbg(querypic)><a href=query_pic.php>查询图片</a></td></tr></table>'
```

```
/*朋友圈管理下拉菜单*/
var myfriend='<table width=90><tr><td id=friendadd onMouseOver=overbg(friendadd) onMouseOut=outbg (friendadd)><a href=friend.php>添加到朋友圈</a></td></tr>\
<tr><td id=browse_fri onMouseOver=overbg(browse_fri) onMouseOut=outbg(browse_fri)><a href=browse_ fri.php>浏览我的朋友</a></td></tr>\
<tr><td id=cxfriend onMouseOver=overbg(cxfriend) onMouseOut=outbg(cxfriend)><a href=query_friend.php>查询朋友信息</a></td></tr></table>'
/*管理员管理下拉菜单*/
var myuser='<table width=90><tr><td id=queryuser onMouseOver=overbg(queryuser) onMouseOut=outbg(queryuser)><a href=queryuser.php>查询用户信息</a></td></tr>\
<tr><td id=browseuser onMouseOver=overbg(browseuser) onMouseOut=outbg(browseuser)><a href=browseuser.php>浏览用户信息</a></td></tr>\
<tr><td id=managepub onMouseOver=overbg(managepub) onMouseOut=outbg(managepub)><a href=managepub.php> 公告管理</a></td></tr>\
</table>'
```

7.10 登录验证码技术专题

当今网络安全越来越受到人们的重视，开发一个带验证码的用户登录模块可以提高网站的安全性。其中，验证码是随机生成的，可以是字母、数字或汉字，也可以是图片。本节技术专题主要就是学习一下验证码的几种实现方法及技巧。

7.10.1 简单的数字验证

先来看一个简单的数字验证码：生成一组 4 位的随机数字，每刷新一次，显示的结果都不相同。如果验证码输入错误，将弹出提示窗口。数字验证码运行结果如图 7.31 所示。



图 7.31 简单的数字验证运行结果

简单的数字验证码在开发过程中主要应用以下两个函数。

（1）mt_rand()函数

mt_rand()函数主要用于从指定参数中随机取一个数值。语法如下：

```
int mt_rand([int min], [int max]);
```

例如，从 52～79 之间取一个随机数值。代码如下：

```
mt_rand(52,79)
```

（2）intval()函数

intval()函数主要用于将变量转成整数类型。语法如下：

```
int intval(mixed var, int [base]);
```

intval()函数可将变量转成整数类型。可省略的参数 base 是转换的基数，默认值为 10。转换的变量 var 可以为数组或类之外的任何类型变量。



7.10.2　数字图形验证码

数字图形验证码相对于数字验证码而言，要稍复杂一些。数字图形验证码主要应用 mt_rand()函数来初始化一组 4 位的随机数，然后利用 for 循环语句将随机生成的 4 位随机验证码利用数字图形进行输出。图形验证码的运行结果如图 7.32 所示。



图 7.32　数字图形验证码

实现数字图形验证码的关键代码如下：

```
<!--js 脚本，判断文本框中输入的验证码是否匹配-->
<script language="javascript">
	function check(form){
		if(form.txt_yan.value==""){
			alert("请输入验证码");form.txt_yan.focus();return false;
		}
		if(form.txt_yan.value!=form.txt_hyan.value){
			alert("对不起，您输入的验证码不正确!");form.txt_yan.focus();return false;
		}
	}
</script>
…
<!--验证表单-->
<form name="form" method="post" action="" onSubmit="return check(form);">
<!--验证码文本框-->
<input type="text" name="txt_yan">
<!--生成随机数字图形验证码-->
<?php
<!--生成随机 4 位数-->
	$num=intval(mt_rand(1000,9999));
<!--使用 for 循环，输出随机 4 位数的图片格式-->
	for($i=0;$i<4;$i++){
		echo "<img src=images/checkcode/".substr(strval($num),$i,1).".gif>";
	}?>
<!--将随机数存入隐藏域-->
<input type="hidden" name="txt_hyan" id="txt_hyan" value="<?php echo $num;?>">
<br> <br>
<input type="submit" name="Submit" value="验证">   <input type="reset" name="Submit2" value="重置">
<!--处理代码段-->
<?php
/*判断验证码录入框是否为空*/
	if($_POST[txt_yan]!=""){
/*当输入的验证码和隐藏域中的验证码相等时，输出欢迎信息*/
		if($_POST[txt_yan] == $_POST[txt_hyan]){
			echo "您输入的验证码通过，感谢您的加盟...";
	}}
?>
```

7.10.3 汉字图形验证码

汉字图形验证码是目前网络上比较流行的验证码方式，这种验证码的实现有一定的难度。首先将图片所对应的汉字依次存储到数组中。然后利用 Rand()函数生成一个小于 10 的数，用于得到随机字符的位置，进而得出随机字符，并设置要显示的随机图片。由于产生的随机字符串是 4 位的，所以需要使用 for 循环语句，根据数组得出相应图片的路径，再将图片转换成数组中的文字，最后输出到浏览器中。

以汉字图形化显示验证码的完整代码如下：

```
<?php
$str=array("大","更","创","天","科","客","博","技","立","新");
$word=strlen($str);
for($i=0;$i<4;$i++){
	$num=rand(0,$word*2-1);
	$img=$img."<img src=' images/checkcode/".$num.".gif' width='16' height='16'>"; //显示随机图片
	$pic=$pic.$str[$num];                                                  //将图片转换成数组中的文字
}
?>
```

7.11 本 章 总 结

本章的博客管理系统首先介绍了博客的基本概念、发展前景、影响范围及博客网的功能分类，使读者对当今主流博客有了一个大致的认识。其次，实现了一个博客系统，其中包含所有基本功能的项目开发，使读者对如何开发一个博客网站有了一个初步的了解。最后，希望读者通过自己的努力，来逐步完善和加强这个博客网的实用功能，最终达到一个令自己满意的作品。

第 8 章

online 影视 365 网

（Apache+PHP+ADODB+phpMyAdmin+MySQL 5.0 实现）

在全球知识经济和信息化高速发展的今天，网络化是企业发展的趋势，21 世纪的人更习惯在网站上听音乐、看电影，所以企业要在同领域中得到突飞猛进的发展，就必须借助网络。

当今社会进入了一个信息快速发展的社会，在网络世界浏览信息的同时也可以听听歌曲，这样既愉悦了身心，又得到了最新的市场动态，由此网络上也出现了很多的影视网站，都很受到欢迎。未来视听生活的新空间，也必然在宽带互联网上开启。VOD 的概念已经被越来越多的人所接受，逐渐成为网络发展的必然趋势之一。通过阅读本章，可以学习到：

- online 影视 365 类网站的整体设计思路及实现方法
- 应用<embed>和<object>播放影音文件
- 关闭子窗口时自动刷新父窗口
- 制作无边框的窗口
- ADODB 介绍及使用
- 更改数据库时的注意事项

8.1 开发背景

×××影视有限公司是一家以影视为主的公司。为了扩大企业规模，增强企业的竞争力，决定向多元化发展。计算机技术、网络通信技术、多媒体技术及数据存储技术的飞速发展，对人类的生产和生活方式产生了很大影响，网络传播以其特有的快速、高效、便捷的传输方式直接被人们所接受。随着多媒体数据的存储、传输和应用技术的不断成熟，以及宽带网络的不断发展，我们有理由相信宽带点播一定会成为网络内容创新的重头戏。

8.2 需求分析

根据客户要求及成本核算，本系统所要实现的功能如下：

☑ 网站实行会员等级管理，分“普通会员”和“高级会员”两种。
☑ 网站提供下载功能和在线视听功能，不同的会员等级实现的功能操作也不同。
☑ 网站分前台和后台两部分，没有权限，用户无法进入后台进行管理。
☑ 独特的点歌模块，为网站的会员提供点歌平台。
☑ 前期数据量比较少，可以使用小型数据库以节约成本，如果后期数据量增大，则可以更换大型数据库。

8.3 系统分析

8.3.1 系统目标

online 影视 365 网主要实现以下目标：

☑ 自定义的鼠标样式增强了网站界面的和谐性。
☑ 采用无边框窗口技术，保障系统的安全性。
☑ 信息搜索灵活、方便。
☑ 采用多媒体引用技术播放影音文件。
☑ 网站采用文件上传和下载技术实现图像、音频、视频文件的客户端上传和下载。

8.3.2 系统功能结构

根据 online 影视 365 网的特点，可以将其分为前台和后台两个部分设计。前台主要实现在线视听、影视音乐上传、影视音乐下载、在线聊天和在线点歌等功能；后台主要用于管理员对影视音乐目录、数据信息和上传日志进行管理。

online 影视 365 网的前台功能结构如图 8.1 所示。



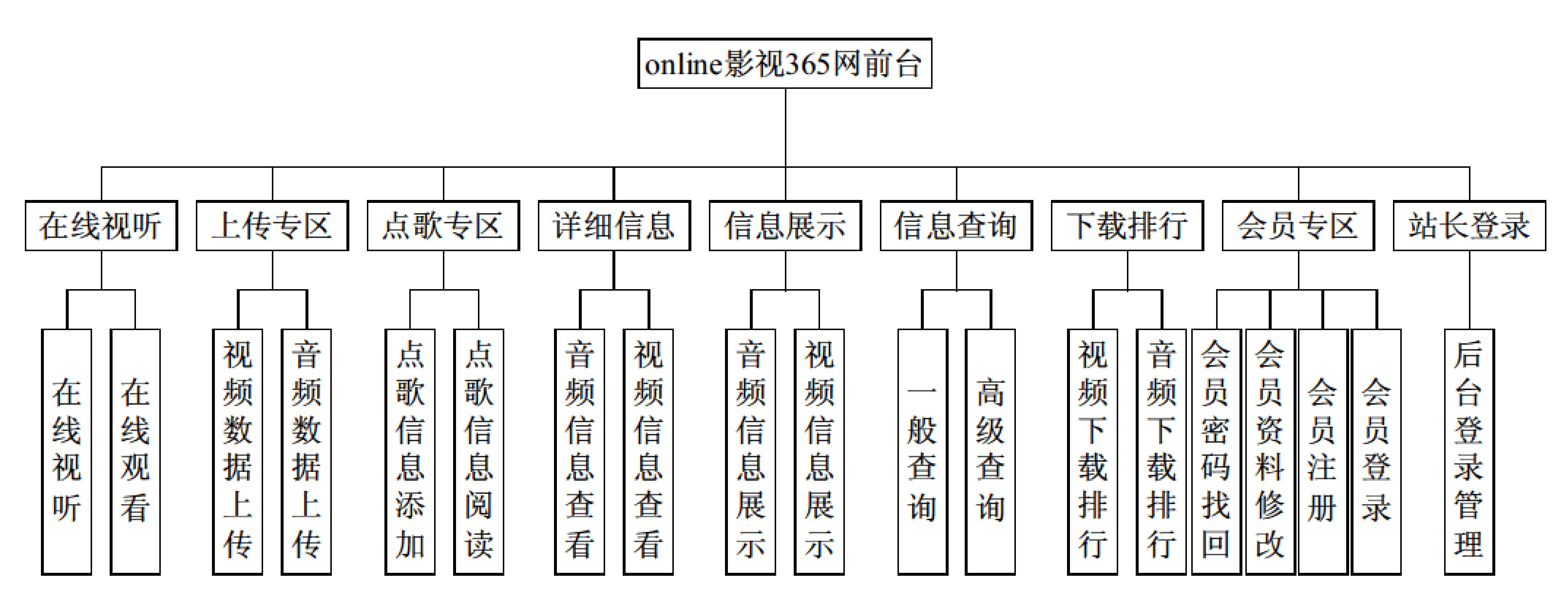


图 8.1　online 影视 365 网前台功能结构图

online 影视 365 网的后台功能结构如图 8.2 所示。

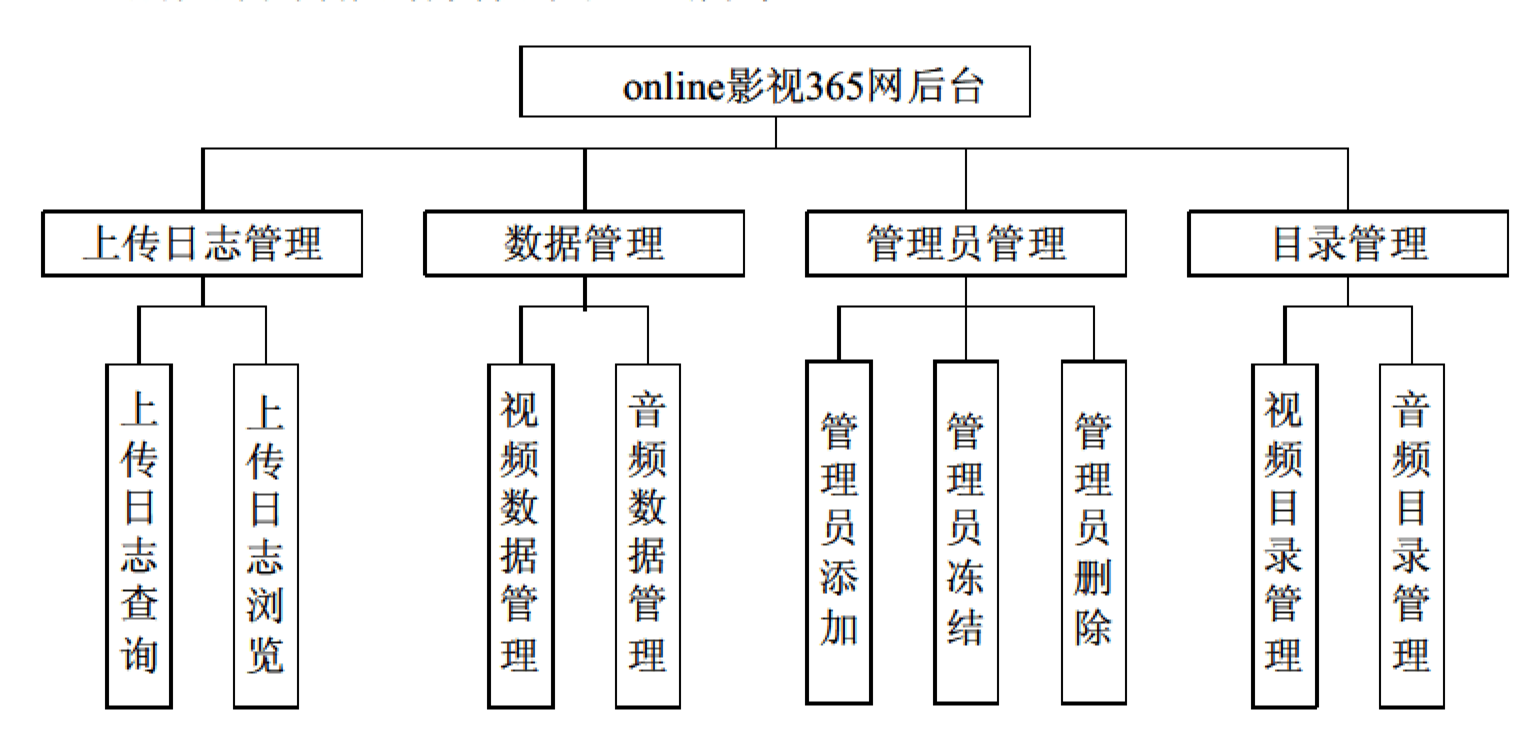
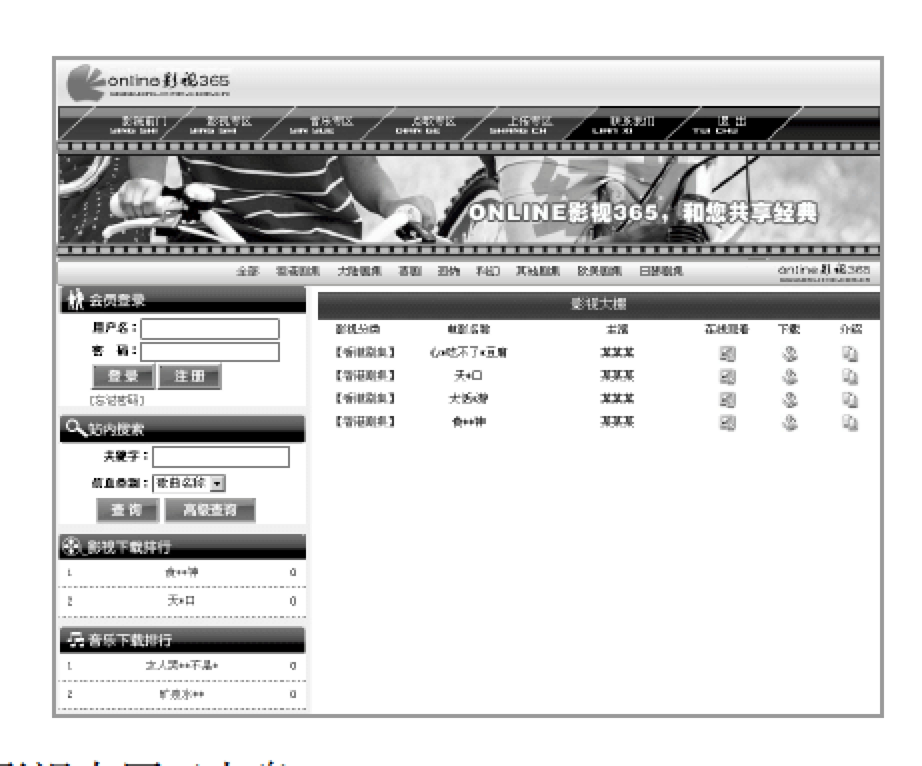


图 8.2　online 影视 365 网后台功能结构图

8.3.3　功能预览

onlie 影视 365 网由多个功能模块组成，为了让读者对本系统有个初步的了解和认识，下面列出几个典型功能的页面，其他页面参见光盘中的源程序。

影视专区页面如图 8.3 所示，该页面展示所有的视频信息及其分类。音乐专区页面如图 8.4 所示，该页面展示所有的音频信息及其分类。



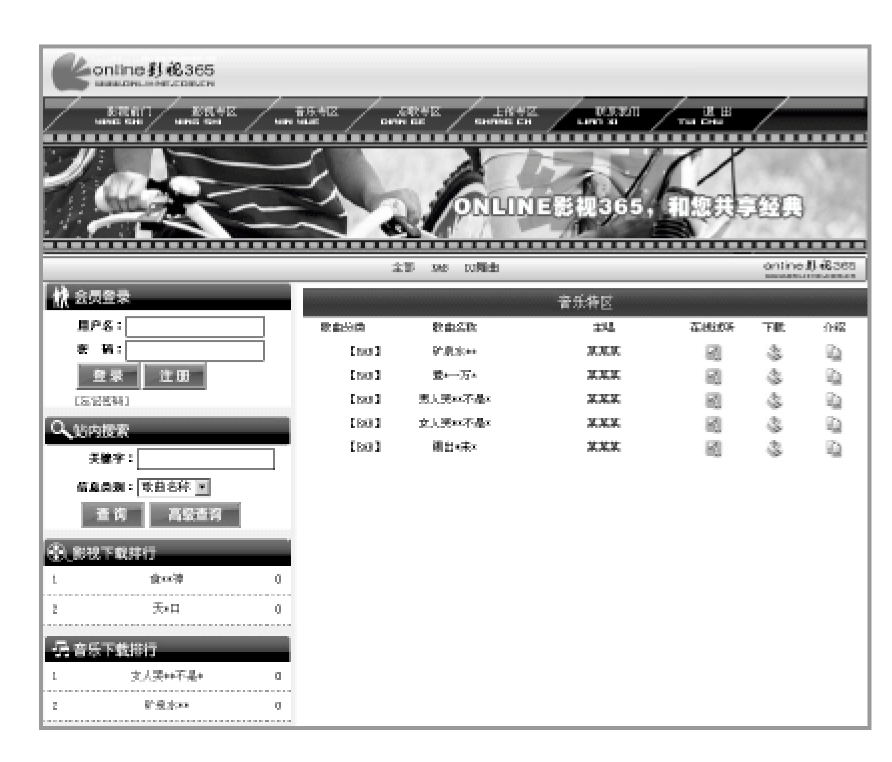


图 8.3　影视专区（光盘\TM\08\online\list.php?action=audio）图 8.4　音乐专区（光盘\TM\08\online\list.php?action=video）

最新影视页面如图 8.5 所示，该页面用于显示所有的最新上传的影视文件。注册页面如图 8.6 所示，该页面显示了游客注册时需要填写的昵称、密码和个人资料等相关内容。

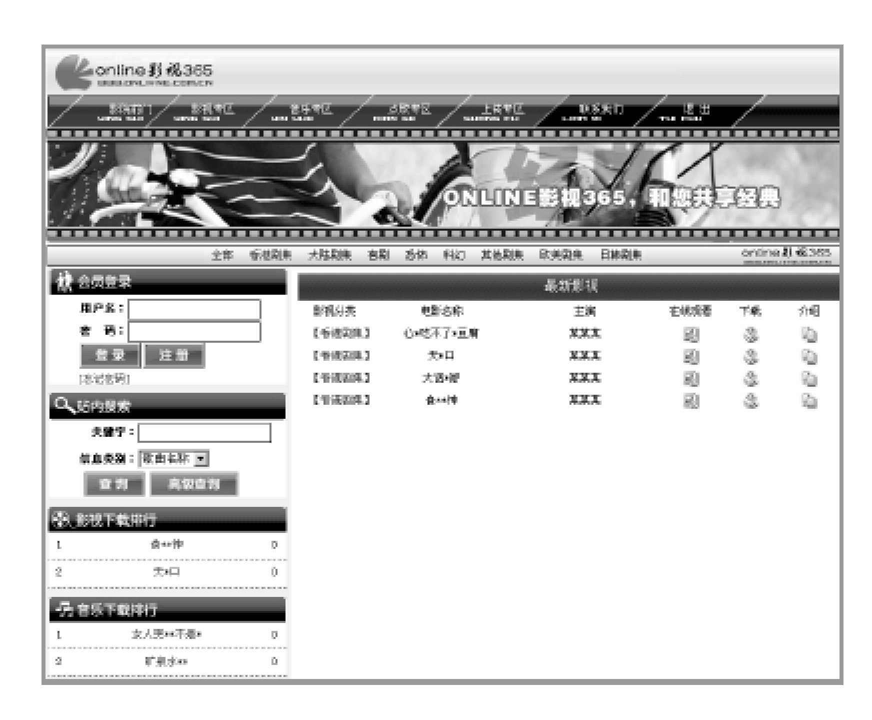


图 8.5　最新影视（光盘\TM\08\online\ list.php）

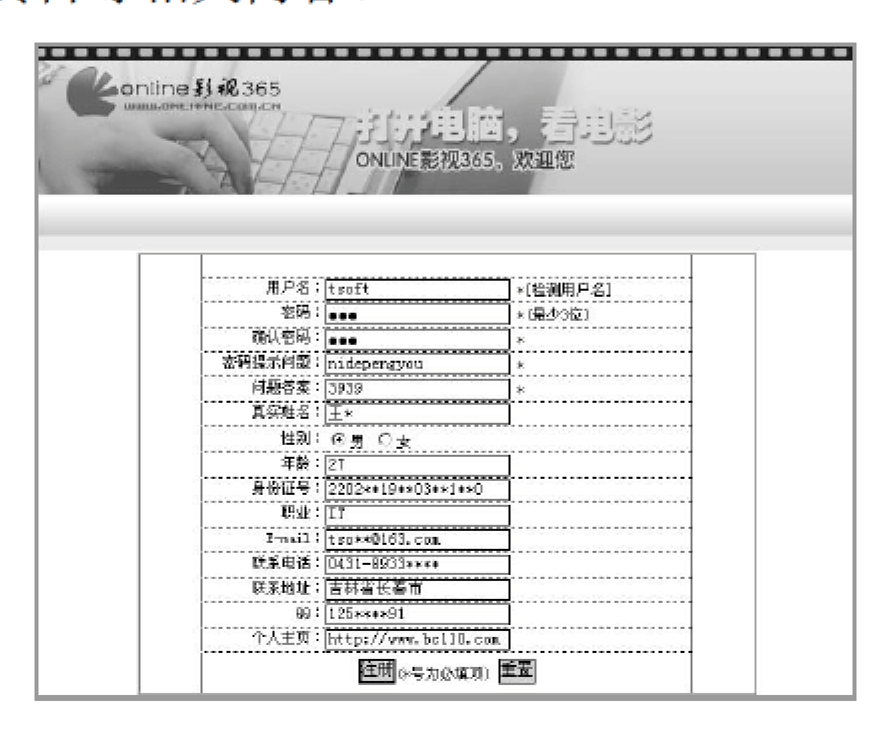


图 8.6　注册（光盘\TM\08\online\register.php）

简单查询页面如图 8.7 所示，该页面显示简单查询所查到的音频、视频文件列表。高级查询页面如图 8.8 所示，该页面用于模糊查询音、视频文件，包含了多个字段，可任意选择。

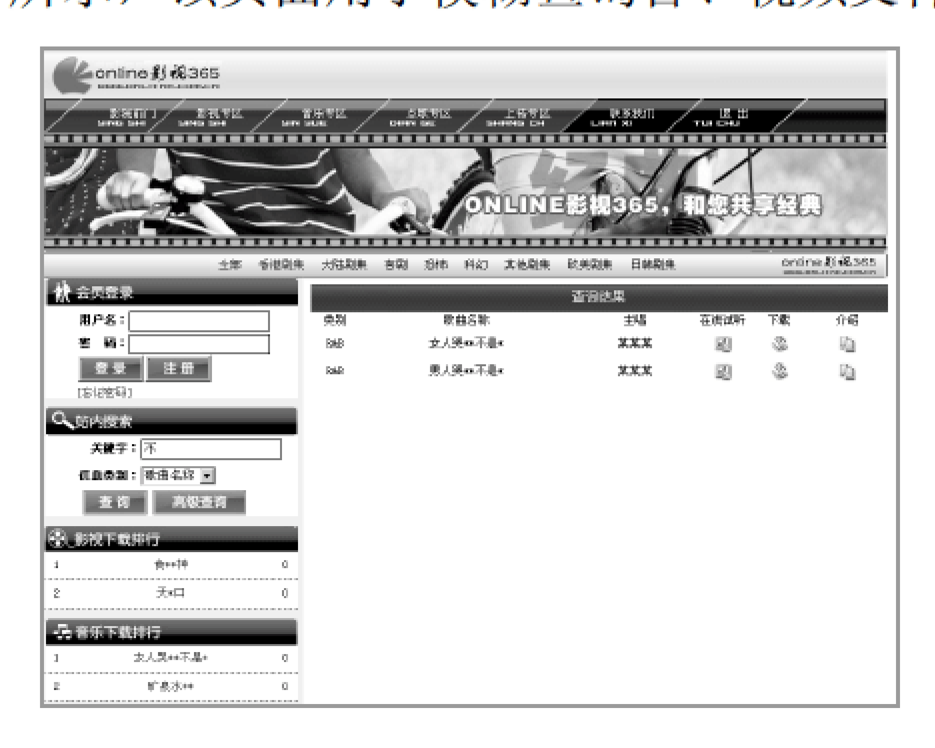


图 8.7　简单查询（光盘\TM\08\online\show.php）

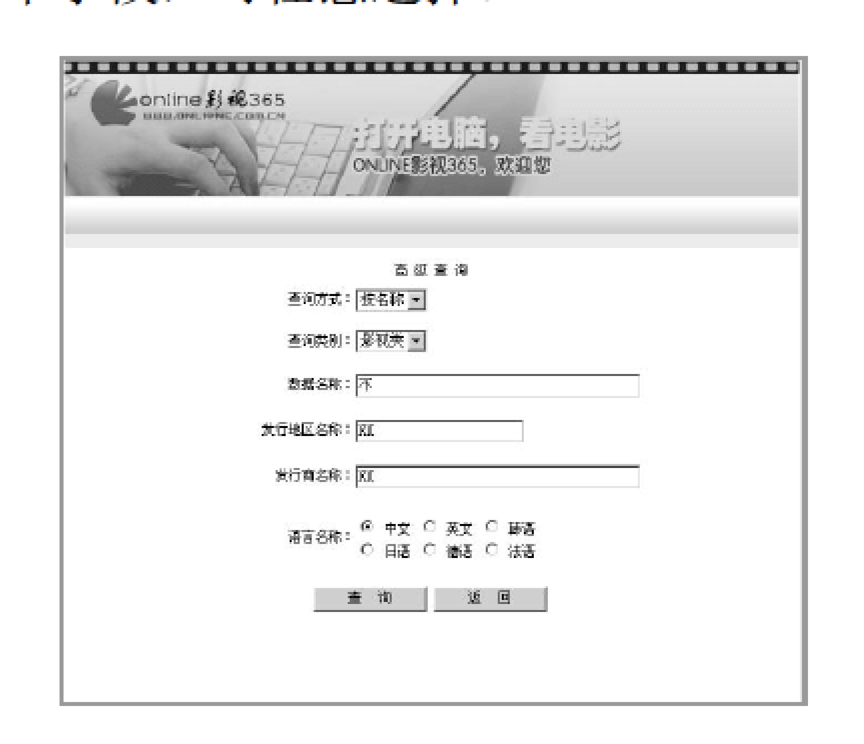


图 8.8　高级查询（光盘\TM\08\online\high.php）

8.3.4　系统流程图

online 影视 365 网的系统流程图如图 8.9 所示。

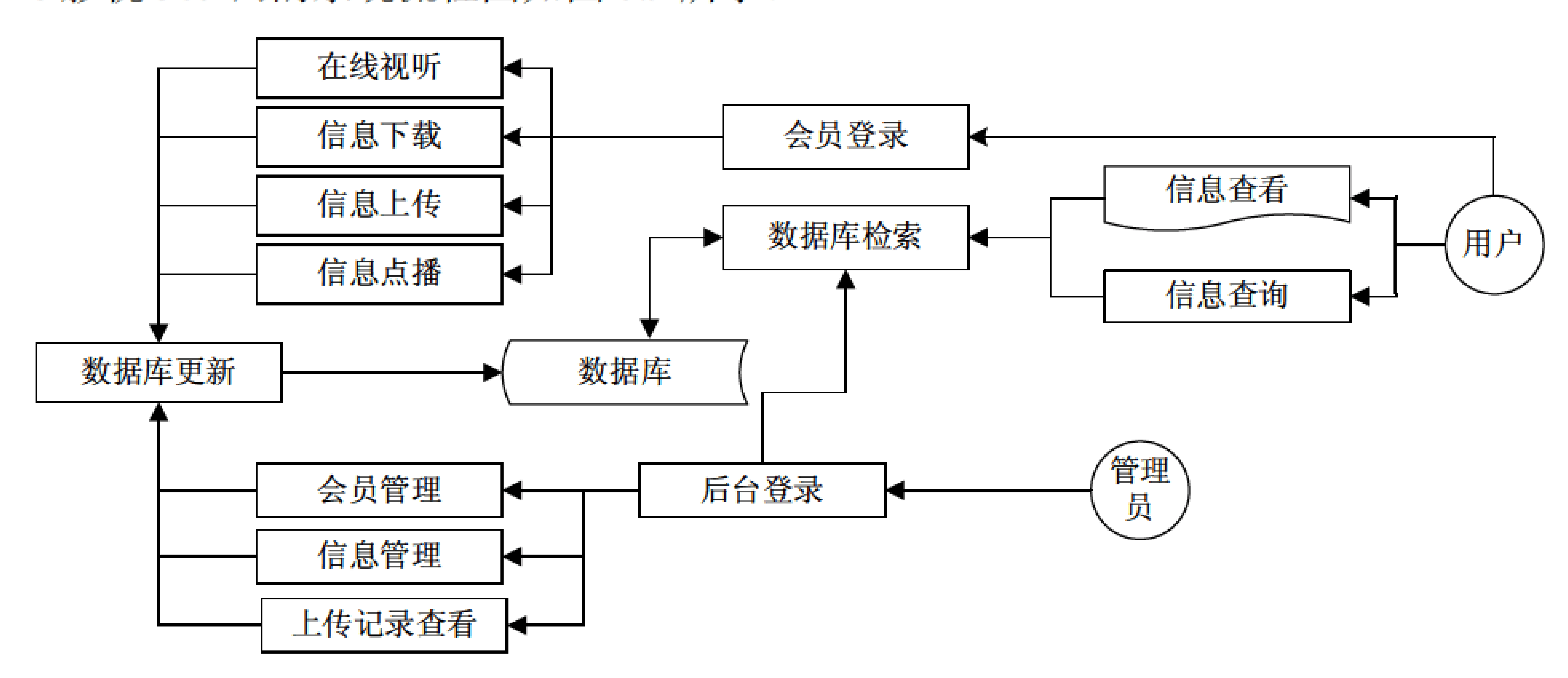


图 8.9　online 影视 365 网的系统流程图



8.3.5　开发环境

在开发 online 影视 365 网时，该项目使用的软件开发环境如下。

1．服务器端

- ☑ 操作系统：Windows Server 2003。
- ☑ 服务器：Apache 2.2.8。
- ☑ PHP 软件：PHP 5.2.6。
- ☑ 数据库：MySQL 5.0.51。
- ☑ MySQL 图形化管理软件：phpMyAdmin-2.10.3。
- ☑ 开发工具：Dreamweaver 8。
- ☑ 浏览器：IE 6.0。
- ☑ 分辨率：最佳效果为 1024×768 像素。

2．客户端

- ☑ 浏览器：推荐使用 IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

8.3.6　文件夹组织结构

online 影视 365 网包括前台和后台两部分，所以文件夹结构也主要由两部分组成。online 影视 365 网的文件夹组织结构图如图 8.10 所示。

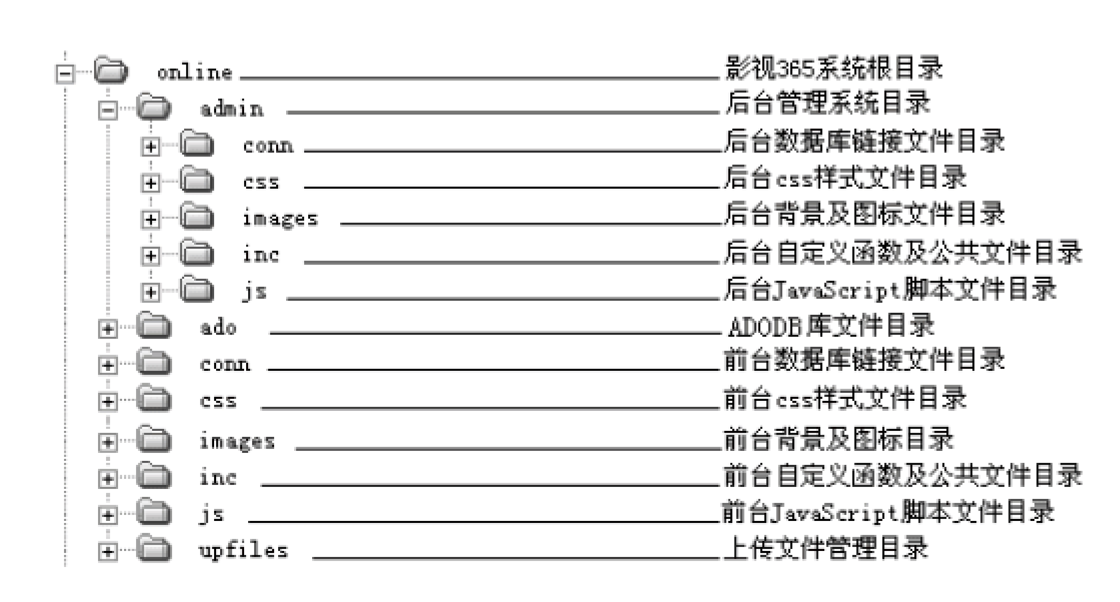


图 8.10　online 影视 365 网的文件夹组织结构图

8.4　数据库设计

8.4.1　数据库分析

本系统使用的是 MySQL 数据库，但使用的是 ADO 连接方式，这是为了便于以后当数据负担加重时，只需要更改其他数据库（如 Oracle），而无须重新编写源程序。

8.4.2　数据库概念设计

通过需求分析和功能上的设计，本系统规划出管理员信息实体、会员信息实体、视频信息实体、音频信息实体、视频目录实体和音频目录实体。下面给出主要的实体及 E-R 图。

会员信息实体包括注册用户的详细个人信息，如果想下载或在线视听，则必须注册为会员才可以。会员信息实体 E-R 图如图 8.11 所示。

视频信息实体包括视频名称、视频图片、视频文件、主要演员、导演、发行商等多项资料。视频信息实体 E-R 图如图 8.12 所示。

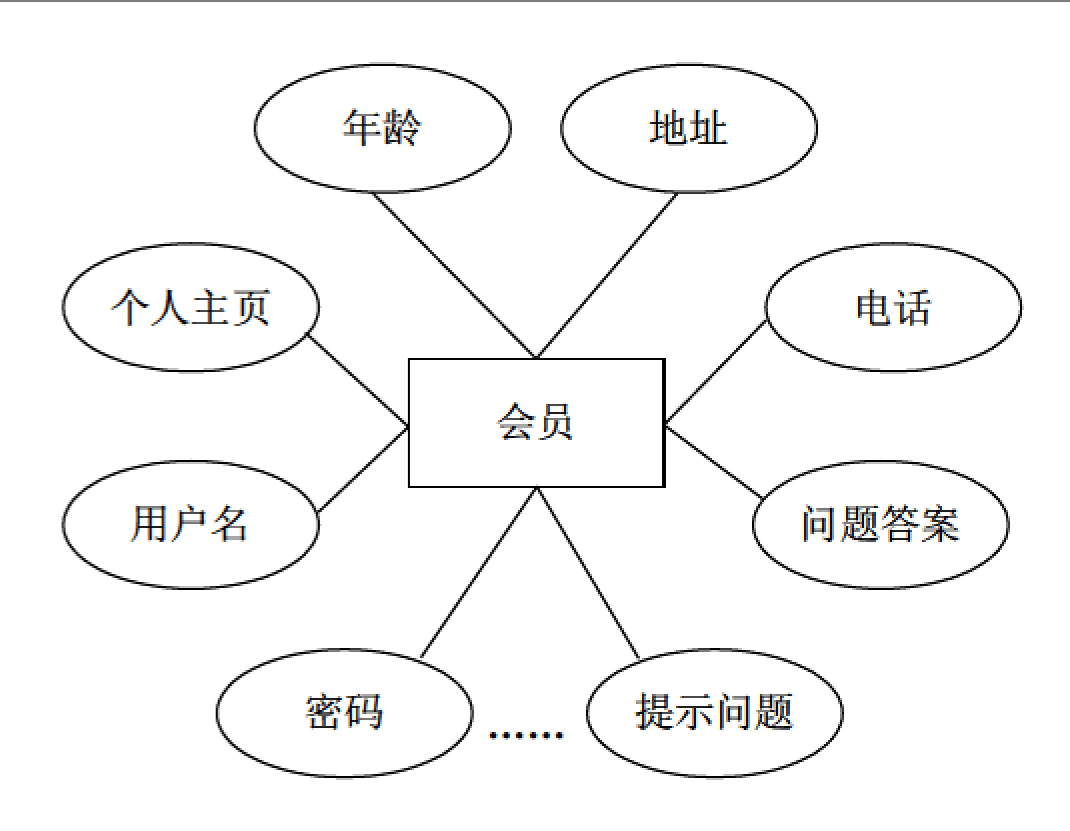


图 8.11　会员信息实体 E-R 图

音频目录实体包括音频目录分类的相关信息，如目录名称、目录级别、父目录名称等。音频目录实体 E-R 图如图 8.13 所示。

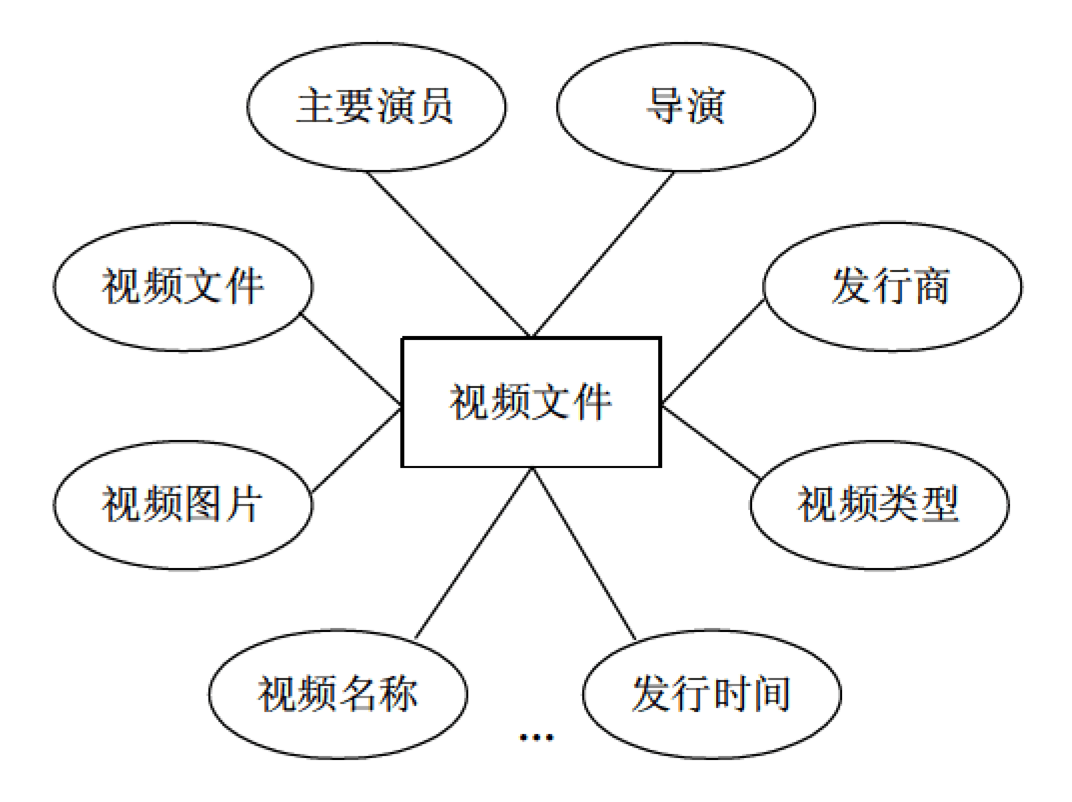


图 8.12　视频信息实体 E-R 图

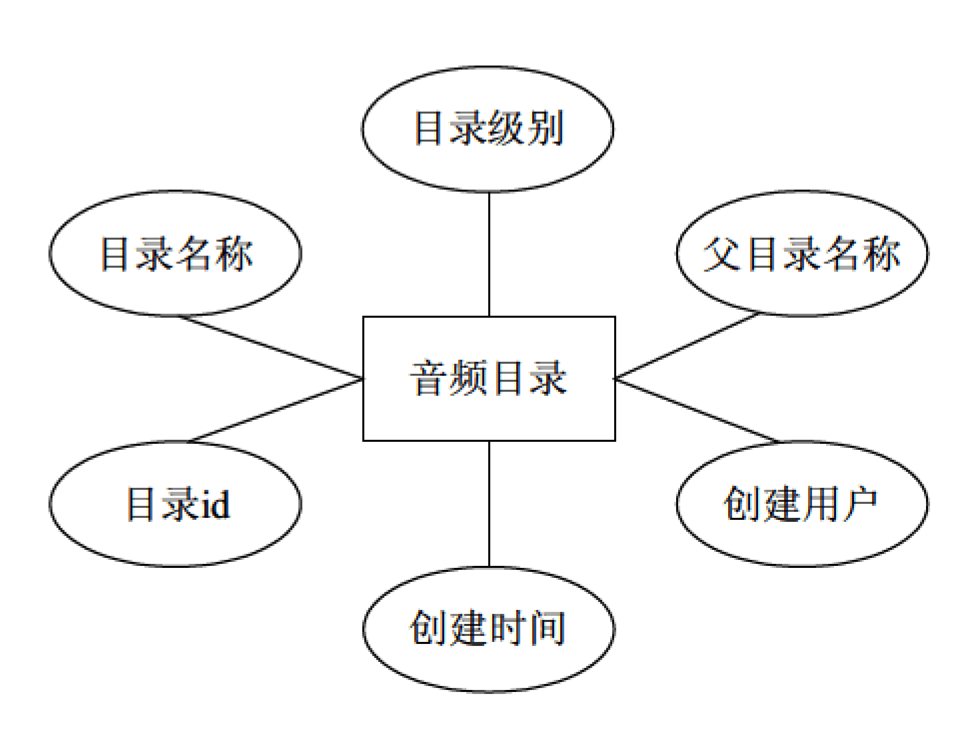


图 8.13　音频目录实体 E-R 图

8.4.3　数据库物理结构设计

online 影视 365 网的数据库中共包含 8 个数据表，如图 8.14 所示。

下面来看一下各个表的结构设计。



服务器: localhost ▸ 数据库: db_online

表	类型	整理	说明
tb_account	MyISAM	gb2312_chinese_ci	会员信息列表
tb_audio	MyISAM	gb2312_chinese_ci	视频信息列表
tb_audiolist	MyISAM	gb2312_chinese_ci	视频目录列表
tb_grade	MyISAM	gb2312_chinese_ci	等级限定列表
tb_manager	MyISAM	gb2312_chinese_ci	管理员列表
tb_register	MyISAM	gb2312_chinese_ci	点歌信息列表
tb_video	MyISAM	gb2312_chinese_ci	音频信息列表
tb_videolist	MyISAM	gb2312_chinese_ci	音频目录列表

图 8.14　db_online 库中的数据库列表

1．tb_account（会员信息列表）

会员信息列表主要存储用户的个人信息，tb_account 表的结构如图 8.15 所示。

2．tb_audio（视频信息列表）

视频信息列表主要上传视频的资料，如视频名称、视频图片等。tb_audio 表的结构如图 8.16 所示。

3．tb_audiolist（视频目录列表）

视频目录列表主要是上传视频所属的类型（二级目录）及类别（一级目录）。tb_audiolist 表的结构



如图 8.17 所示。

服务器: localhost ▸ 数据库: db_online ▸ 表 : tb_account

字段	类型	整理	属性	Null	默认	说明
id	int(11)			否		自动编号
name	varchar(30)	gb2312_chinese_ci		否		会员昵称
password	varchar(30)	gb2312_chinese_ci		否		会员密码
question	varchar(50)	gb2312_chinese_ci		否		密码提示问题
answer	varchar(50)	gb2312_chinese_ci		否		密码提示答案
realName	varchar(30)	gb2312_chinese_ci		否		会员真实姓名
numbers	varchar(20)	gb2312_chinese_ci		否		身份证号
sex	varchar(10)	gb2312_chinese_ci		否		会员性别
age	int(4)			否		会员年龄
job	varchar(50)	gb2312_chinese_ci		否		所做工作
email	varchar(100)	gb2312_chinese_ci		否		E-mail
address	varchar(100)	gb2312_chinese_ci		否		联系地址
phone	varchar(20)	gb2312_chinese_ci		否		联系电话
qq	varchar(20)	gb2312_chinese_ci		否		联系QQ
http	varchar(100)	gb2312_chinese_ci		否		个人主页
counts	int(4)			否	0	上传次数
grade	varchar(10)	gb2312_chinese_ci		否	普通会员	会员级别
whether	varchar(10)	gb2312_chinese_ci		否	1	是否激活

图 8.15　会员信息列表结构

服务器: localhost ▸ 数据库: db_online ▸ 表 : tb_audio

字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
name	varchar(100)	gb2312_chinese_ci		否		视频名称
picture	varchar(200)	gb2312_chinese_ci		否		封面图片
sizes	varchar(50)	gb2312_chinese_ci		否		视频大小
grade	varchar(10)	gb2312_chinese_ci		否		视频级别
publisher	varchar(100)	gb2312_chinese_ci		否		发行公司
actor	varchar(200)	gb2312_chinese_ci		否		主要演员
director	varchar(100)	gb2312_chinese_ci		否		导演
marker	varchar(100)	gb2312_chinese_ci		否		制片
languages	varchar(20)	gb2312_chinese_ci		否		语种
type	varchar(50)	gb2312_chinese_ci		否		类型
style	varchar(50)	gb2312_chinese_ci		否		类别
froms	varchar(100)	gb2312_chinese_ci		否		视频所属地区
publishTime	date			否		发行时间
remark	varchar(1000)	gb2312_chinese_ci		否		视频介绍
property	varchar(20)	gb2312_chinese_ci		否		视频属性
address	varchar(200)	gb2312_chinese_ci		否		存储地址
username	varchar(50)	gb2312_chinese_ci		否		发布人姓名
issueDate	datetime			否		发布时间
downTime	int(4)			否	0	下载次数
lastTime	datetime			否		最后下载时间
whether	varchar(20)	gb2312_chinese_ci		否	1	是否为新品

图 8.16　视频信息列表结构

4．tb_manager（管理员列表）

管理员列表主要用于后台管理员的资料信息管理，包括管理员名称、管理员权限等。tb_manager 表的结构如图 8.18 所示。

服务器: localhost ▸ 数据库: db_online ▸ 表 : tb_audiolist

字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
grade	varchar(20)	gb2312_chinese_ci		否		目录级别
name	varchar(50)	gb2312_chinese_ci		否		目录名称
father	varchar(50)	gb2312_chinese_ci		否		父级目录
userName	varchar(30)	gb2312_chinese_ci		否		创建人昵称
issueDate	datetime			否		创建时间

图 8.17　视频目录列表结构

服务器: localhost ▸ 数据库: db_online ▸ 表 : tb_manager

字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
name	varchar(50)	gb2312_chinese_ci		否		管理员昵称
password	varchar(50)	gb2312_chinese_ci		否		管理员密码
type	varchar(50)	gb2312_chinese_ci		否		管理员类型
realName	varchar(50)	gb2312_chinese_ci		否		真实姓名
issueDate	datetime			否		创建时间
whether	varchar(50)	gb2312_chinese_ci		否		是否激活

图 8.18　管理员列表结构

5．tb_register（点歌信息列表）

点歌信息列表主要存储会员用户之间的点歌祝福的信息，主要包括发送人、接收人、歌曲地址等。tb_register 表的结构如图 8.19 所示。

服务器: localhost ▸ 数据库: db_online ▸ 表 : tb_register

字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
name	int(4)			否		歌曲地址
fromName	varchar(50)	gb2312_chinese_ci		否		发送人姓名
toName	varchar(50)	gb2312_chinese_ci		否		接收人姓名
remark	varchar(50)	gb2312_chinese_ci		否		备注
issueDate	datetime			否		点歌时间

图 8.19　点歌信息列表结构

6．tb_video（音频信息列表）

音频信息列表主要用于音频文件的信息管理，如歌手、歌词等。tb_video 表的结构如图 8.20 所示。

7．tb_videolist（音频目录列表）

音频目录列表主要用于音频文件的分类管理，有目录名称、目录级别等。tb_videolist 表的结构如图 8.21 所示。

服务器: localhost ▸ 数据库: db_online ▸ 表 : tb_video						
字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
name	varchar(100)	gb2312_chinese_ci		否		音频名称
picture	varchar(200)	gb2312_chinese_ci		否		封面图片
actor	varchar(100)	gb2312_chinese_ci		否		主唱
ci	varchar(50)	gb2312_chinese_ci		否		歌词作者
qu	varchar(50)	gb2312_chinese_ci		否		歌曲作者
actortype	varchar(50)	gb2312_chinese_ci		否		歌手类型
style	varchar(50)	gb2312_chinese_ci		否		音频类别
publisher	varchar(100)	gb2312_chinese_ci		否		发行商
froms	varchar(100)	gb2312_chinese_ci		否		发行地区
type	varchar(50)	gb2312_chinese_ci		否		音频类型
sizes	varchar(50)	gb2312_chinese_ci		否		音频大小
languages	varchar(20)	gb2312_chinese_ci		否		音频语言
publishTime	date			否		发行时间
remark	varchar(1000)	gb2312_chinese_ci		否		备注
property	varchar(20)	gb2312_chinese_ci		否		音频属性
address	varchar(200)	gb2312_chinese_ci		否		音频地址
userName	varchar(50)	gb2312_chinese_ci		否		发布人姓名
issueDate	datetime			否		发布时间
downTime	int(4)			否		下载次数
lastTime	datetime			否		最后下载时间
whether	varchar(50)	gb2312_chinese_ci		否	1	是否为新品

图 8.20　音频信息列表结构

服务器: localhost ▸ 数据库: db_online ▸ 表 : tb_videolist						
字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
grade	varchar(50)	gb2312_chinese_ci		否		目录等级
name	varchar(50)	gb2312_chinese_ci		否		目录名称
father	varchar(50)	gb2312_chinese_ci		否		父级目录
userName	varchar(50)	gb2312_chinese_ci		否		创建人姓名
issueDate	datetime			否		创建时间

图 8.21　音频目录列表结构

8.5　前台首页设计

8.5.1　前台首页概述

online 影视 365 网的前台首页的功能模块主要包括以下 3 部分。

☑　网站首部导航栏：主要有影视专区、音乐专区、上传专区和点歌专区。

☑　网站左侧导航栏：包括会员登录模块、搜索模块和影视排行模块。

☑　网站主显示区：包括最新上传模块、视频模块和音频模块。

本案例中提供的前台首页如图 8.22 所示。该首页在本书光盘中的路径为\TM\08\online\index.php。



图 8.22　online 影视 365 前台首页



8.5.2　前台首页技术分析

在本章的数据库分析中已经提到，本系统所使用的连接数据库的方式为 ADODB 连接。这里，先来初步了解一下 ADODB 如何连接 MySQL。

下面以本系统使用的数据库和数据表为例，来介绍一下使用 ADODB 的基本操作步骤。

（1）载入（include）adodb.in.php 文件。

```
include "../ado/adodb.inc.php";
```

（2）建立数据库类型连接。

```
$conn = &ADONewConnection("mysq");   or    $conn = &NewADOConnection("mysql");
```

（3）连接要使用的数据库。

```
$conn->PConnect("db_online");       or    $conn->Connect("db_online");
```

（4）执行 SQL 语句。

```
$recordset = $conn->Execute("select * from tb_audio");
```

（5）判断结果，并对结果处理。

```
if($recordset)                                      //判断 execute()函数执行中是否有错误
    while(!$recordset->EOF){                        //如果没有错误，则配合 wihle 语句循环输出结果
        echo $recordset->fields[0]."<br>";
        $recordset->movenext();                     //指针下移
    }
}
```

（6）关闭连接。

```
$recordset->close();
$conn->close();
```

如果是对微软的 ADO 有所了解的用户，会对上述步骤有一种亲切感。无论从字段名称，还是从功能上，两者都非常相似，所以上手会很轻松。没有接触过 ADO 的用户也不用心急，用 ADODB 操作数据库主要就是这 6 个步骤，只要记住这个顺序，做几遍练习后，也会运用自如。

关于 ADODB 类库的详细讲解请参见 8.14 节。

8.5.3　前台首页的实现过程

前台首页使用的数据表：tb_account、tb_audio、tb_video

在首页中，使用到了 3 个 include 语句，将导航栏、登录框、搜索框、浏览区等主要的模块加载进来。限于篇幅，这些页面的代码不做讲解，请自行到光盘中查找。前台首页的实现代码如下：

例程01 代码位置：光盘\TM\08\online\index.php

```
<?php
      session_start();                                           //开启 session 支持
?>
<script src="js/chk.js" language="javascript"></script>         //引入 js 脚本文件
<link rel="stylesheet" href="css/style.css" />                   //引入 css 样式文件
<?php
      include "top.php";                                         //首部导航及 LOGO
?>
<table border="0" cellpadding="0" cellspacing="0">
  <tr><td width="265" align="center" valign="top">
<?php
      include "left.php";                                        //登录、搜索框
?>
      </td><td width="605" align="center" valign="top">
<?php
      include "main.php";                                        //主浏览区
?>
      </td></tr></table>
```

8.6 详细信息查看模块

8.6.1 信息查看模块概述

详细信息查看模块主要包括视频详细信息查看和音频详细信息查看两个部分。查看信息功能没有权限限制，无论是游客还是会员都可以进行查看。但查看页面中的下载和在线视听按钮则只有会员才可以看到，游客只能看到“返回”按钮，用来关闭查看的页面。信息查看模块的框架图如图 8.23 所示。

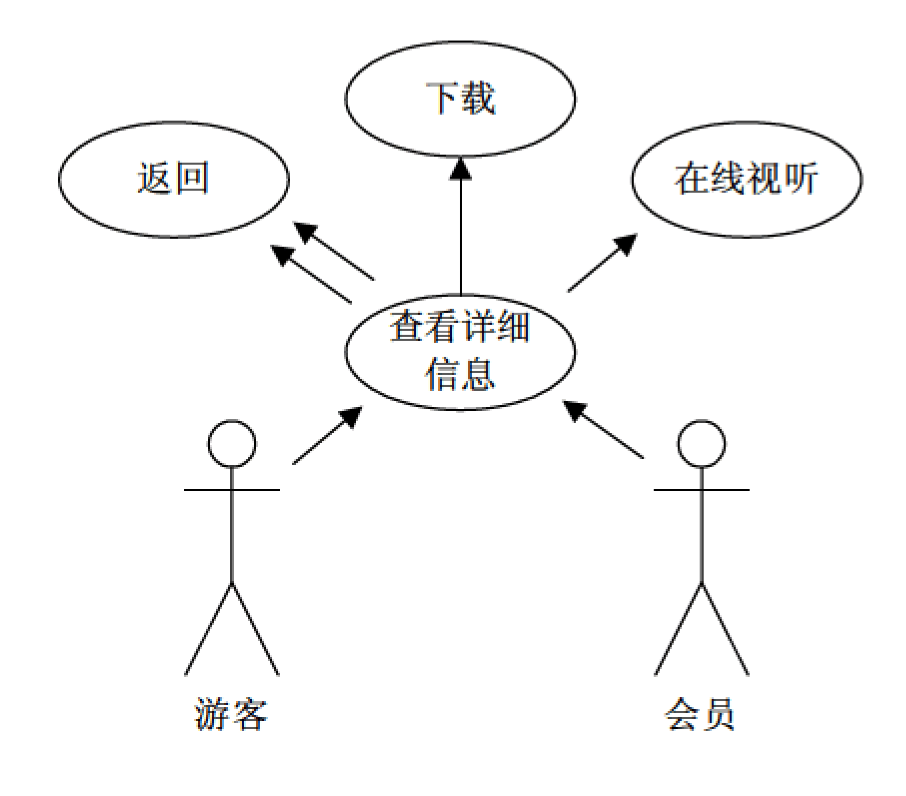
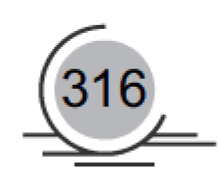


图 8.23　信息查看框架图

8.6.2 信息查看模块技术分析

在这个模块中，所要解决的技术难点就是：系统需要根据不同的浏览用户，显示不同的操作权限。这里是使用 if 语句，配合 session 来实现其功能的。首先通过检测$_SESSION[name]是否存在，来判断





用户是否登录，如果没有登录，则跳过“在线视听”和“下载”按钮，只显示“返回”按钮；如果用户为登录会员，则显示“在线播放”按钮。接着判断$_SESSION[grade]是否为高级会员，如果登录会员为普通会员，则“下载”按钮为灰色失效状态；如果登录会员为高级会员，则同时显示 3 个按钮的全部功能。该功能的关键代码如下：

例程02　代码位置：光盘\TM\08\online\intro.php

```
<?php
    /*如果用户是登录会员*/
     if($_SESSION[name]<>"" ){
?>
    <!--显示“播放”按钮-->
❶ <input name="Submit" type="submit" value="  播  放  " onclick="javascript:Wopen=open ('operation.php?action=see&id=<?php echo $rst->fields[16]; ?>','','height=700,width=665,scrollbars=no');">
<?php
    /*如果是登录会员*/
    }if($_SESSION[name]<>""){
?>
    /*如果是高级会员，则“下载”按钮为激活状态，否则为灰色*/
❷  <input name="Submit" type="button" <?php  if ($_SESSION[grade]<>"高级会员"){ echo "disabled";}?> value="  下  载  " onclick="javascript:Wopen=location='download.php?action=audio&id=<?php echo $rst->fields[16]; ?>';">
<?php
    }}
?>
/*显示“返回”按钮*/
<input name="Submit2" type="button" value="  返  回  " class="submit"
❸  onClick="javascript:top.window.close();"></td>
```

代码贴士

❶ open()方法可以打开一个无边框的子窗口。关于 open()方法的介绍，可以参考 8.13 节。

❷ disabled：译为无效的、不起作用的，这里是 button 按钮的属性，即不可点击状态。

❸ top.window.close()：关闭弹出窗口。

8.6.3　信息查看的实现过程

消息管理模块使用的数据表：tb_audio、tb_video

用户可以在信息展示页面单击图标进入详细信息展示页面。详细信息展示页面主要是根据传递的参数进行数据库检索，并将结果集输出到浏览器中。查看页面的运行结果如图 8.24 所示。



图 8.24　详细信息页面的运行结果

在详细信息展示页面（intro.php）中，通过 id 值从数据表中提取所要显示的信息输出到页面，当文件信息输出完毕后，通过 session 值来判断访问者是游客还是会员，如果是游客，则只显示“返回”按钮；如果是会员，还要显示“在线试听”按钮和“下载”按钮。程序的关键代码如下：

例程03　代码位置：光盘\TM\08\online\intro.php

```
<?php
        session_start();                                        //开启 session 支持
        include "conn/conn.php";                                //载入数据库连接文件
?>
<link rel="stylesheet" href="css/style.css">                    //引入 css 样式文件
<!--显示数据详细信息-->
…
<?php    $sql="select * from tb_audio where id=".$_GET[id];    //根据 id 生成查询语句
            $rst = $conn->execute($sql);                        //执行 SQL 语句，返回结果集
/*如果结果集不为空，则输出信息*/
if(!$rst->EOF){
?>
        <table width="400" border="0" align="center" cellpadding="0" cellspacing="0">
            <tr>
            <td height="15" colspan="2"> </td>
            </tr>
            <!--输出视频名称-->
            <tr>
              <td width="131" height="20" align="right" valign="middle">名称：</td>
              <td width="269" height="20"><?php echo $rst->fields[1]; ?></td>
            </tr>
            <!--输出文件大小-->
            <tr>
              <td height="20" align="right" valign="middle">大小：</td>
              <td height="20"><?php echo $rst->fields[3]; ?></td>
            </tr>
             …
            <tr>
              <td height="30" colspan="2" align="center" valign="middle">
            /*根据不同用户，显示不同权限*/

…
```

8.7　点歌模块设计

8.7.1　点歌模块概述

用户通过主导航条进入点歌专区。在点歌专区，可以对专区内的音乐进行试听、浏览音频信息的详细内容和打开点歌页面进行点歌，进行点歌的前提条件是用户必须登录。点歌模块的用例图如图 8.25 所示。

图 8.25　点歌模块的用例图

8.7.2　点歌模块技术分析

本系统的在线点歌模块实现了会员间发送祝福与点歌的功能，收到祝福的会员可以实现在线听歌。

本节和 8.8 节的在线视听模块中，将逐一介绍实现在线视听的两种方法。本节先来介绍 HTML 标签中的播放多媒体标签<embed>。

1．基本语法

```
<embed src=url>
```

url 为音频或视频文件，其路径可以是相对路径，也可以是绝对路径。

<embed>可以用来播放各种多媒体文件，格式可以是 midi、wav、mp3 等，当下主流的浏览器都支持该标签。

2．属性设置

<embed>标签有很多属性，常用的属性及说明如表 8.1 所示。

表 8.1　<embed>常用属性及说明

属　性　名	说　　明	举　　例
autostart	该属性规定音频或视频文件是否在下载完之后就自动播放。 true 表示音乐下载完成后自动播放；false 表示下载完成后不播放	<embed src="1.mp3" autostart=true>
loop	该属性规定音频或视频文件是否循环及循环次数。 属性值为 true 时，音频或视频文件循环；属性值为 false 时，音频或视频文件不循环。如果为正整数，则为循环次数	<embed src="1.mp3" loop=true>
hidden	该属性规定控制面板是否显示，默认值为 no。 true 表示隐藏面板；no 表示显示面板	<embed src="1.mp3" hidden="yes">
starttime	该属性规定音频或视频文件开始播放的时间。未定义则从文件开头播放	<embed src="1.mp3" starttime="00:10">
volume	该属性规定音频或视频文件的音量大小。未定义则使用系统本身的设定	<embed src="1.mp3" volume="10">
width	该属性规定了控制面板的宽度	<embed src="1.mp3" width="100">
height	该属性规定了控制面板的高度	<embed src="1.mp3" height="200">
title	该属性规定了音频或视频文件的说明文字	<embed src="1.mp3" title="a good song">

除了这些属性外，<embed>标签还可以对面板的外观及其背景色和前景色进行设置，感兴趣的读者可以查找相关的资料。

8.7.3　点歌模块的实现过程

点歌模块使用的数据表：tb_register、tb_video

单击“点歌专区”超链接，首先进入到点歌专区页面，页面中的音乐试听和信息查看在其他模块中都有介绍，这里主要看一下点歌功能。单击小图标，将进入到点歌信息页面，点歌页面的运行结果如图 8.26 所示。

在点歌信息页面，需要用户添加祝福语和接收人名称。点歌页面涉及的 HTML 表单元素如表 8.2 所示。

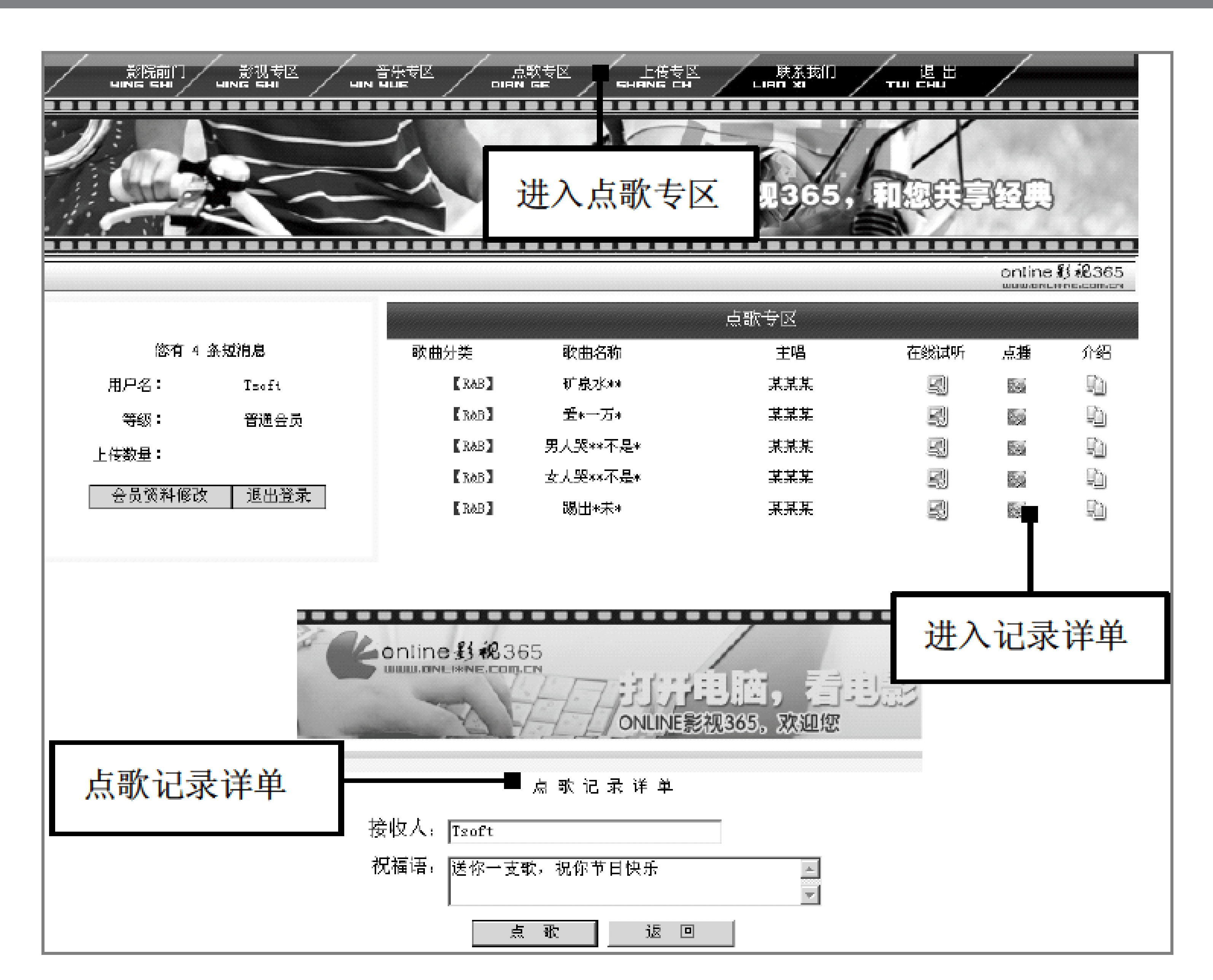


图 8.26　点歌模块的运行结果

表 8.2　点歌页面涉及的 HTML 表单元素

名　　称	类　　型	重 要 属 性	含　　义
list	form	action="Register.asp?action=add"	表单
toname	text	id="toname" size="30"	接收人名称
remark	textarea	cols="40" rows="5"　id="remark"	简要介绍
id	Hidden	value="<?php echo $_GET[id] ?>"	所点歌曲的地址
submit	button	onClick="register()"	“点歌”按钮
submit2	button	onClick="javascript:top.window.close()"	“返回”按钮

在点歌页面，将前页传过来的歌曲 id、接收人、祝福语等信息组成 insert 语句存储到数据库中，不管保存成功与否，都将关闭当前子窗口，回到父窗口中进行操作。程序代码如下：

例程04　代码位置：光盘\TM\08\online\give.php

```
<?php
    session_start();                                    //开启 session 支持
    …
    if($_POST[toname] <> ""){                           //如果接收人不为空
    $id=$_POST[id];                                     //歌曲 id
    $toname=$_POST[toname];                             //接收人姓名
    $from=$_SESSION[name];                              //发送人姓名
    $remark=$_POST[remark];                             //备注信息
    <!--生成 insert 语句-->
    $sql="insert into tb_register Values('','".$id."','".$from."','".$toname."','".$remark."','".date("Y-m-d H:i:s")."')";
    $rst = $conn->execute($sql);                        //执行 SQL 语句
?>
<script language="javascript">
```

```
<?php
	if(!($rst == false)){
?>
	alert("点歌信息保存成功");
<?php
	}else{
?>
	alert("点歌失败");
<?php
}
?>
	top.window.close();
</script>
<?php
}
?>
```

8.8　在线视听模块设计

8.8.1　在线视听模块概述

在线视听主要包括视频数据在线观看和音频数据在线收听两个部分。在不同的页面都可以直接进入视听页面，包括首页、搜索结果页、内容介绍页和列表页。但必须是登录会员才可以，游客是没有权限的。在线视听流程图如图 8.27 所示。

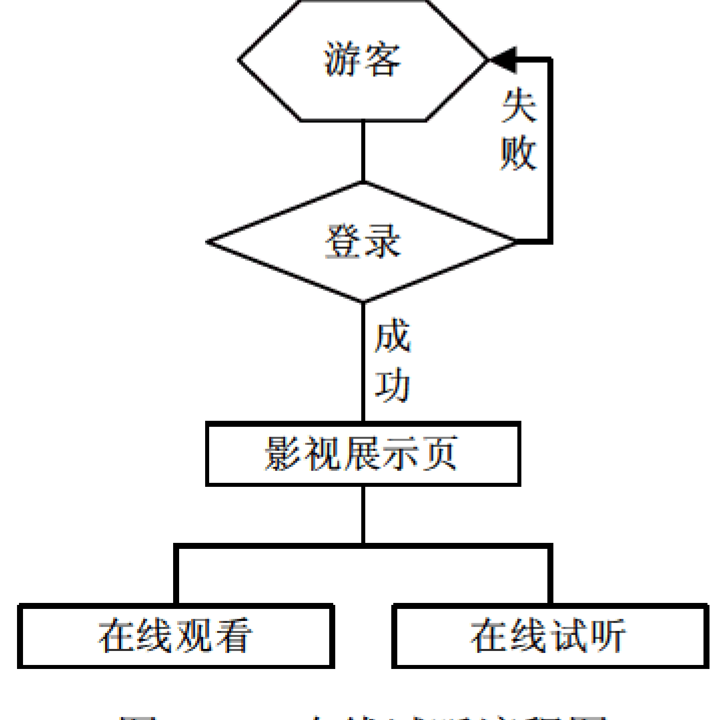


图 8.27　在线试听流程图

8.8.2　在线视听模块技术分析

在 8.7.2 节中学习了如何使用<embed>多媒体标签来播放影音文件。本节来学习另一种方法：<object>标签。

使用<object>标签来定义一个嵌入的对象。该对象向 HTML 页面添加多媒体。此元素运行插入到 HTML 页面中对象的数据和参数，以及可用来显示和操作数据的代码。<object>标签的基本格式如下：

```
<object classid="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95" height="68" id="MediaPlayer1" width="460">
  <param name="ShowStatusBar" value="-1">
  <param name="Filename" value="<?php echo $_GET[id]?>">
</object>
```

☑ classid：定义嵌入 Windows Registry 中或某个 URL 中类的 ID 值，此属性可用来指定浏览器中包含的对象的位置。

☑ param：可定义用于对象的 run-time 设置。

<object>是微软专门为 IE 浏览器打造的、可以扩展外部应用程序及插件的对象标签，它和<embed>



的不同之处在于：<object>只支持以 IE 技术为核心的浏览器系列，对其他的浏览器则无效。而且如果要使用<object>来播放多媒体，那么需要安装相应的播放插件，如 RealPlayer、QuickTime 等。

8.8.3 在线视听模块的实现过程

在线视听模块使用的数据表：tb_audio、tb_video

会员可以在首页、信息展示页面或者详细信息展示页面单击“观看”或“试听”按钮进行在线视听操作。页面运行结果如图 8.28 所示。



图 8.28　在线视听模块的运行结果

以视频详细信息展示页面为例，用户可以单击 图标，打开视频播放页面（see.php）进行在线观看。视频播放页面主要是根据传递的参数进行数据库检索，并将对应数据的硬盘存储地址作为多媒体文件的引用地址，当页面加载完毕后，视频数据将自动播放，实现在线观看功能。在线播放页面的主要代码如下：

例程05　代码位置：光盘\TM\08\online\see.php

```
<?php
    session_start();                                    //开启 session 支持
?>
<body>
    <!--使用 object 对象播放视频/音频文件-->
<object classid="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95" width="665" height="500" id="MediaPlayer1" >
    <!--设置自动播放-->
<param name="AutoStart" value="-1">
    <!--设置状态栏-->
  <param name="ShowStatusBar" value="-1">
    <!--播放文件路径-->
  <param name="Filename" value="upfiles/audio/<?php echo $_GET[id];   ?>">
</object>
</body>
```



8.9　后台首页设计

8.9.1　后台首页概述

根据用户对各个功能模块的使用频率和重要程度，本系统的首页面中要显示的模块主要有两部分，分别介绍如下。

（1）网站左侧导航栏：包括各个管理模块及分类。

☑　目录管理模块：主要包括视频目录管理和音频目录管理两个部分。

☑　数据管理模块：主要包括视频数据管理和音频数据管理两个部分。

☑　会员管理模块：包括会员等级管理、会员信息冻结、会员信息删除 3 个部分。

☑　上传信息管理模块：主要用于浏览和查询用户上传操作的详细内容。

☑　管理员设置模块：主要包括管理员信息添加、删除、冻结 3 个部分。

（2）网站主显示区：显示各个模块的操作及结果。

本案例中提供的后台首页如图 8.29 所示。该首页在本书光盘中的路径为\TM\08\online\admin\ index.php。



图 8.29　online 影视 365 网后台首页

8.9.2　后台首页技术分析

本系统的后台首页采用的布局结构为二分栏布局。左侧导航栏清楚、明白地显示了后台管理员所能使用到的功能。当单击任意功能按钮时，在主显示区显示对应的操作界面和该功能模块下的子功能。页面简练、结构清晰、浏览方便，二分栏的主要特点被表现得淋漓尽致。关于二分栏的详细介绍，请参见 3.6.3 节。

8.9.3　后台首页的实现过程

后台首页使用的数据表：tb_manager

下面来看一下后台首页的实现过程。在首页代码中，对每项功能权限的判断都是在 left.php 中完成

的，只要在左侧导航栏的位置载入 left.php 即可。在主显示区，根据参数返回的不同值，显示不同的功能界面。程序的关键代码如下：

例程06　代码位置：光盘\TM\08\online\admin\main.php

```
<?php
	session_start();                                                        //开启 session 支持
	include "inc/chec.php";                                                 //载入权限检查文件
	include "conn/conn.php";                                                //载入数据库链接文件
?>
<link href="css/style.css" rel="stylesheet"/>                               //引入 css 样式文件
<script src="js/admin_js.js" language="javascript"></script>                //引入 js 脚本文件
<center>
…
<?php
/*载入网站左侧导航栏*/
	include "left.php";
?>
…
<div style="height:35px;"> </div>
<?php
/*根据不同的参数，显示不同的功能界面*/
	if(isset($_GET[action])){
		switch ($_GET[action]){
			case "audioList":                                               //视频目录添加
				include "a_list.php";
				break;
			case "videoList":                                               //音频目录添加
				include "v_list.php";
				break;
			case "audio":                                                   //视频数据添加
				include "audio.php";
				break;
			case "video":                                                   //音频数据添加
				include "video.php";
				break;
			case "grade":                                                   //会员等级设置
				include "grade.php";
				break;
			case "member";                                                  //会员数据设置
				include "member.php";
				break;
			case "log";                                                     //上传日志管理
				include "log.php";
				break;
			case "manager";                                                 //管理员设置
				include "manager.php";
				break;
		}}
?>			</td></tr>
		<tr><td height="15" bgcolor="#f0f0f0"> </td></tr>
	</table>
```



```
        </td></tr>
        </table>
</center>
```

8.10　目录管理模块设计

8.10.1　目录管理模块概述

目录管理模块主要包括视频目录管理和音频目录管理两个部分。管理员可以通过后台管理导航进入对应的目录管理页面。

如果管理员进入了视频目录管理页面（audiolist.php），那么在该页面内可以打开目录添加页面进行目录添加操作，也可以删除相应的目录。鉴于目录信息比较简短，本例中没有提供目录修改的功能。目录管理模块的用例图如图 8.30 所示。

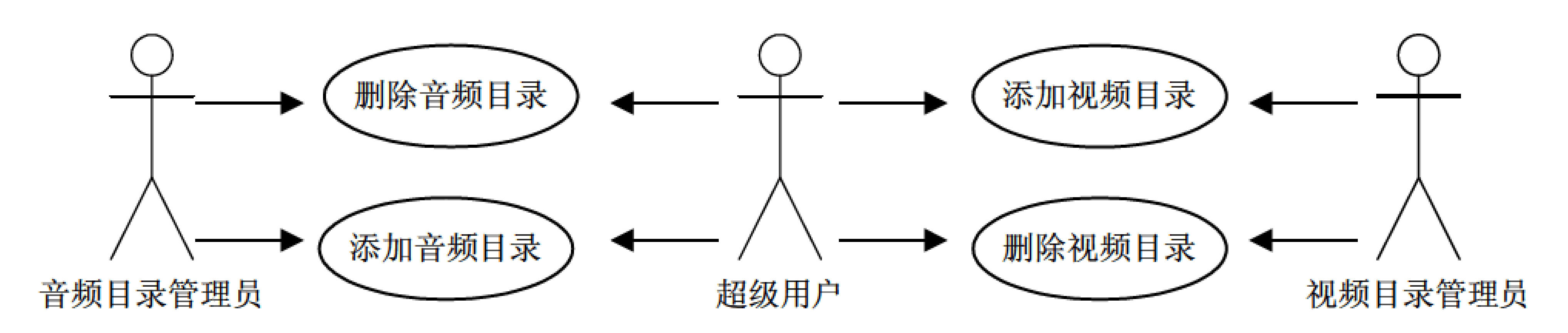


图 8.30　视频目录管理用例图

说明　由于视频目录管理和音频目录管理实现的方法基本类似，因此，本章重点讲解视频目录管理模块实现的方法，关于音频目录管理模块的实现方法请参见本书附赠源码光盘。

8.10.2　目录管理模块技术分析

在本系统中，有个功能经常要被用到：就是在添加目录、歌曲时，经常会碰到重名的问题。对于这个问题，可以写一个自定义函数，并将它放到单独的一个文件中，方便其他页面调用，如本系统中的/inc/func.php，就是专门用来存储自定义函数的文件。

下面来看一下自定义函数 is_chk()。

例程07　代码位置：光盘\TM\08\online\admin\inc\func.php

```
//判断目录名是否重复
//$f_fields：字段名
//$tablename：数据表名
//$f_str：要查找的字段
function is_chk($f_fields,$tablename,$f_str){
        $conn = &ADONewConnection('mysql');                          //建立 MySQL 连接
        $conn->PConnect("localhost","root","root","db_online");      //连接 db_online 数据库
        $is_chk = true;                                              //声明一个 boolean 型变量
        $is_sqlstr = "select $f_fields from $tablename";             //查找字段
```

```
        $is_rst = $conn->execute($is_sqlstr);
        while(!$is_rst->EOF){                                          //循环输出记录集
        /***********——对比字段中的值***********/
            if($f_str == $is_rst->fields[0]){
        /*如果字段在数据库中找到，说明重名了，is_chk 设置成 false*/
                $is_chk = false;
                break;
            }
            $is_rst->MoveNext();                                       //将指针指向下一条记录
        }
        return $is_chk;                                                //返回变量$is_chk
}
```

8.10.3 目录管理的实现过程

目录管理使用的数据表：tb_audiolist、tb_videolist

单击左侧导航栏中的“视频目录管理”超链接，打开视频目录管理页面，在该页面中，有一个“目录添加”超链接、所有的一、二级目录信息和对应的“删除”按钮。视频目录管理页面的运行结果如图 8.31 所示。



图 8.31　视频目录管理的运行结果

视频目录管理页面的代码很简单：使用 while 循环将数据库中的记录全部显示出来，并在每输出一条记录后，添加一个“删除”操作。目录管理页面的实现代码如下：

例程08　代码位置：光盘\TM\08\online\admin\a_list.php

```
<?php
        session_start();                                               //开启 session 支持
        include "conn/conn.php";                                       //载入数据库链接文件
        include "inc/chec.php";                                        //载入权限检查文件
                /*输出视频目录数据表中的全部数据*/
        $l_sqlstr = "select * from tb_audiolist";
        $l_rst = $conn->execute($l_sqlstr);
```

```
?>
…
                <tr>
                /*目录添加*/
                  <td height="10" colspan="5" align="right" valign="middle"><a href="#" onclick="javacript:
Wopen= open ('operation.php? action= audiolist','添加目录','height=500,width=665,scrollbars=no');">目
录添加</a></td>
                </tr>
                <!--显示目录相关信息的字段名-->
                <tr>
                  <td height="30" align="center" valign="middle">ID</td>
                  <td height="30" align="center" valign="middle">等级</td>
                  <td height="30" align="center" valign="middle">名称</td>
                  <td height="30" align="center" valign="middle">父级名称</td>
                  <td height="30" align="center" valign="middle">操作</td>
                </tr>
                <?php
                     while(!$l_rst->EOF){
                ?>
                <tr>
                  <!--输出目录 id-->
                  <td height="18" align="center" valign="middle"><?php echo $l_rst->fields[0]; ?></td>
                  <!--输出目录等级-->
                  <td height="18" align="center" valign="middle"><?php echo $l_rst->fields[1]; ?></td>
                  <!--输出目录名称-->
                  <td height="18" align="center" valign="middle"><?php echo $l_rst->fields[2]; ?></td>
                  <!--输出目录父级名称-->
                  <td height="18" align="center" valign="middle"><?php echo $l_rst->fields[3] ?></td>
                  <!--输出“删除”超链接-->
                  <td height="18" align="center" valign="middle"><a href="del_list_chk.php?action= audiolist&id=
<?php echo $l_rst->fields[0]; ?>" onclick="return del_chk();">删除</a>
                  </td>
                </tr>
                <?php
                /*指针下移一条记录*/
                $l_rst->MoveNext();
                }
                ?>
            </table></td>
          </tr>
      </table></td>
  </tr>
</table>
```

8.10.4　目录添加的实现过程

目录添加使用的数据表：tb_audiolist、tb_videolist

单击视频目录管理页面中的“目录添加”按钮，将打开目录添加页面。目录添加页面的运行结果如图 8.32 所示。





图 8.32　视频目录添加的运行结果

视频目录添加页面（audiolist.php）中主要包含一个添加目录表单，表单的主要元素如表 8.3 所示。

表 8.3　视频目录添加页面涉及的 HTML 表单元素

名　　称	类　　型	重 要 属 性	含　　义
list	form	method="post" action="audiolist_chk.php"	视频目录添加表单
grade	select	name="grade" OnChange="check()" <option value="1" selected>一级目录</option> <option value="2">二级目录</option>	目录等级
father	select	<?php $l_sqlstr = "select * from tb_audiolist where grade = '1'"; $l_rst = $conn->execute($l_sqlstr); while(!$l_rst->EOF){ ?> <option value="<?php echo $l_rst->fields[2]; ?>"><?php echo $l_rst->fields[2]; ?></option> <?php $l_rst->MoveNext(); } ?>	父级目录名称
names	text	type="text" id="names"	目录名称
Submit	submit	class="submit" value="添　加" onclick="return n_chk();"	“添加”按钮
Submit2	button	class="submit" value="返　回" onClick="javascript:top.window.close()"	“返回”按钮

当管理员添加信息提交后，系统转到处理页中（audiolist_chk.php）进行添加操作。首先判断输入目录名称是否与已存在的名称重复，如果重复，回到上一步；如果不重复，则向数据表中添加新记录。视频目录添加的程序代码如下：

例程09　代码位置：光盘\TM\08\online\admin\audiolist_chk.php

```
<?php
    session_start();                                        //开启 session 支持
    include "inc/chec.php";                                 //载入权限检查文件
    include "conn/conn.php";                                //载入数据库连接文件
    include "inc/func.php";                                 //载入自定义函数文件
    /*使用自定义函数 is_chk()，判断用户名是否重复*/
    if(is_chk("name","tb_audiolist",$_POST[names]) == false){
```

```
        echo "<script>alert('名称重复');history.go(-1);</script>";          //如果重复，退到上一步操作
        exit();
    }
    /*生成 insert 插入语句*/
    $a_sqlstr = "insert into tb_audiolist (grade,name,father,userName,issueDate)
values('$_POST[grade]','$_POST[names]','$_POST[father]','$_SESSION[admin]','".date("Y-m-d H:i:s")."')";
    /*判断返回结果*/
    if($conn->execute($a_sqlstr) == false)
    /*如果添加失败，返回上一步*/
        echo "<script>alert('添加失败');history.go(-1);</script>";
    else
    /*如果添加成功，则刷新父窗口，关闭子窗口*/
        echo "<script>top.opener.location.reload();alert('添加成功');window.close();</script>";
?>
```

8.10.5　目录删除的实现过程

目录删除使用的数据表：tb_audiolist、tb_videolist

视频目录的删除功能是根据传递的目录 id 号将数据库中对应的记录找到，并删除。程序代码如下：

例程10　代码位置：光盘\TM\08\online\admin\del_list_chk.php

```
<?php
    session_start();                                                  //开启 session 支持
    include "inc/chec.php";                                           //载入权限检查文件
    include "conn/conn.php";                                          //载入数据库链接文件
    if($_GET[action] == "audiolist")                                  //如果 action 等于 audiolist
        $t_name = "tb_audiolist";                                     //要访问的数据表就为 tb_audiolist
    else if($_GET[action] == "videolist")                             //如果 action 等于 videolist
        $t_name = "tb_videolist";                                     //那么数据库表就为 tb_videolist
    $sqlstr = "delete from ".$t_name." where id =".$_GET[id];         //根据 id 号，删除对应记录
    /*如果删除成功，则返回到当前页，否则，退到上一步*/
    if($rst = $conn->execute($sqlstr) == false){
        echo "<script>alert('删除错误！ ".$sqlstr."');history.go(-1);</script>";
    }else
        echo "<script>alert('删除成功');location='".$_SERVER['HTTP_REFERER']."';</script>";
?>
```

8.11　数据管理模块设计

8.11.1　数据管理模块概述

数据管理模块主要包括视频数据管理和音频数据管理两个部分。管理员可以通过后台管理导航进入对应的数据管理页面，如视频数据管理页面（audio.php）。在数据管理页面，用户可以打开数据添加页面或删除对应的数据。数据管理模块用例图如图 8.33 所示。

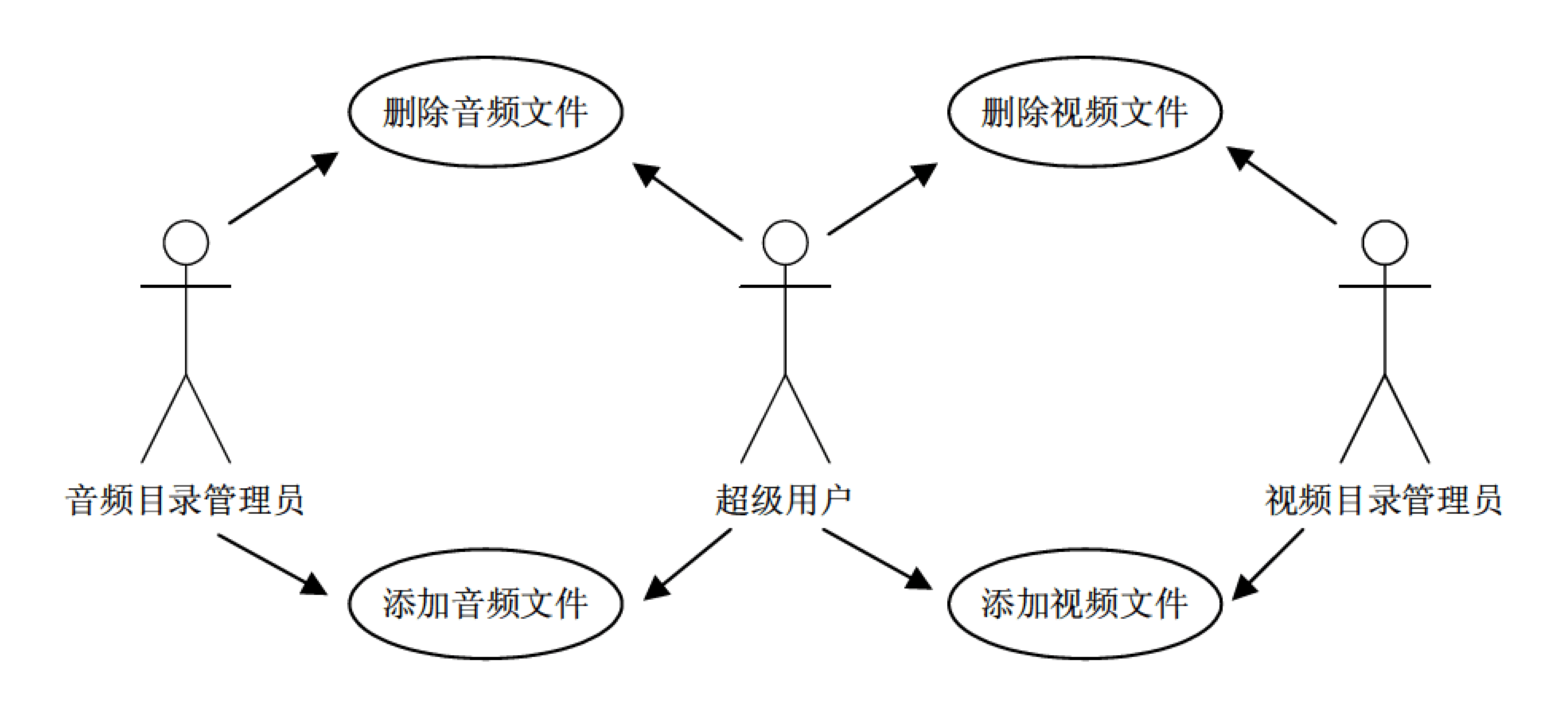


图 8.33　数据管理模块用例图

说明 由于视频数据管理和音频数据管理实现的方法基本类似，因此，本章重点讲解视频数据管理模块实现的方法，关于音频数据管理模块的实现方法请参见本书附赠源码光盘。

8.11.2　数据管理模块技术分析

在本模块中，主要运用的就是文件上传技术。关于文件上传方面的知识，请参见本书 8.7 节，里面有对预定义变量$_FILES 的详尽介绍，和如何判断图片类型和大小的相关方法。

在这个模块中，略有不同的是使用了一个自定义函数来判断文件名后缀。下面来看看这个函数的实现过程，代码如下：

例程11　代码位置：光盘\TM\08\online\admin\inc\func.php

```
//判断文件后缀
//$f_type：允许文件的后缀类型
//$f_upfiles：上传文件名
function f_postfix($f_type,$f_upfiles){
    $is_pass = false;                                  //设置一个 boolean 型变量$is_pass
❶   $tmp_upfiles = split("\.",$f_upfiles);             //拆分上传文件名，存储到变量$tmp_upfiles 中
❷   $tmp_fix = $tmp_upfiles[count($tmp_upfiles) - 1];  //取得上传文件名后缀
    for($num = 0; $num < count($f_type);$num++){       //使用 for 循环，输出允许使用的后缀类型
❸       if(strtolower($tmp_tix) == $f_type[$num])     //如果上传文件后缀包括在其中，说明类型相符
            $is_pass =true;                            //$is_pass 变量赋值为 true
    }
    return $is_pass;                                   //返回变量$is_pass
}
```

代码贴士

❶ split("\.",$f_upfiles)：split()函数为字符分割函数，它有两个参数，分别为分隔符和要分割字符串。返回结果为分割后的字符串数组。例如，$tmp=split("c","placepublicslice")，那么$tmp 值就为$tmp[0]="plac"、$tmp[1]="epubli"、$tmp[3]="sli"、$tmp[4]="e"。

❷ count($tmp_upfiles)：count()函数的作用是取得数组个数。





❸ strtolower($tmp_tix)：strtolower()函数返回参数字符串的小写格式，如 strtolower("AbC")，输出等于 abc。

说明　尽可能地使用 PHP 函数库的函数。因为自己写的函数，无论是从算法设计，还是从功能实现上，都无法和 PHP 自带的函数相比较。不要试着去找寻它们的错误，这是非常不明智的做法。

8.11.3　数据添加的实现过程

数据添加使用的数据表：tb_audio、tb_video

管理员可以通过单击视频数据管理页面的“数据添加”按钮，打开数据添加页面（audio.php），在数据添加页面填写了表单之后，将表单提交给本页，本页会根据提交的表单数据在数据库中添加对应的记录。数据添加页面的运行结果如图 8.34 所示。

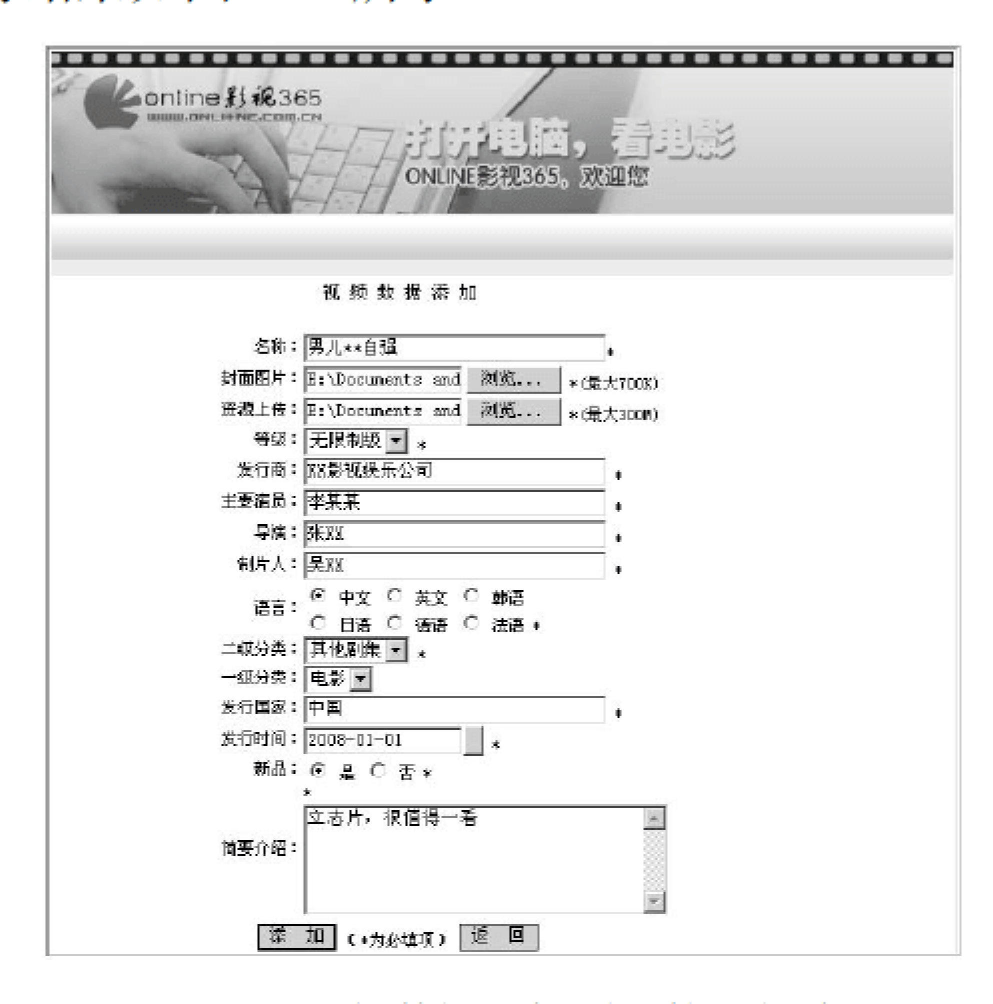


图 8.34　视频数据添加页面的运行结果

在视频数据添加页面中涉及的 HTML 表单元素如表 8.4 所示。

表 8.4　视频数据添加页面涉及的 HTML 表单元素

名　称	类　型	重 要 属 性	含　义
List	form	name="list" method="post" action="dataadd_chk.php" enctype="multipart/form-data"	表单
Names	text	id="names" size="30"	数据名称
Picture	text	id="picture" size="15"	图片名称
Address	text	id="address" size="15"	数据位置
Grade	select	<option value="一级">一级</option> <option value="二级">二级</option> <option value="三级">三级</option> <option value="无限制级" selected="selected">无限制级</option> <option value="禁片">禁片</option>	数据等级
Publisher	text	id="publisher" size="30"	发行商

续表

名　称	类　型	重 要 属 性	含　义
Actor	text	id="actor" size="30"	主要演员
Director	text	id="director" size="30"	导演
Maker	text	id="marker" size="30"	制片人
Language	radio	<input type="radio" name="language" value="中文" checked> 中文 <input type="radio" name="language" value="英文">英文 <input type="radio" name="language" value="韩语">韩语 <input type="radio" name="language" value="日语">日语 <input type="radio" name="language" value="德语">德语 <input type="radio" name="language" value="法语"> 法语 * </td>	语种

Style	select	<?php
		　　$a_sqlstr="select * from tb_videoList where grade='2' ";
		　　$a_rst = $conn->execute($a_sqlstr);
		　　while(!$a_rst->EOF){
		?>
		<option value="<?php echo $a_rst->fields[2]; ?> "><?php echo $a_rst->fields[2]; ?></option>
		<?php
		　　$a_rst->movenext();
		　　}
		?>
			类别

Types	select	<?php
		$t_sql = "select * from tb_audiolist where grade = '1' ";
		$t_rst = $conn->execute($t_sql);
		while(!$t_rst->EOF){
		?>
		<option value="<?php echo $t_rst->fields[2]; ?> "><?php echo $t_rst->fields[2]; ?></option>
		<?php
		$t_rst->movenext();
		}
		?>
			类型

From	text	id="from" size="30"	发行国家
publishtime	text	loadCalendar(list.publishtime);	发行时间
news	radio	<input name="news" type="radio" value="1" checked>是 <input name="news" type="radio" value="0">否	新品参数
remark	textarea	class="submit" onclick="return a_chk();"	简要介绍
Submit	button	class="submit" onclick="return a_chk();"	"添加"按钮
Submit2	button	OnClick="javascript:top.window.close()"	"返回"按钮

视频管理处理页（dataadd_chk.php）主要分 3 部分：图片文件格式及图片大小判断、视/音频格式及文件大小判断，信息都准确无误后，则向数据库中添加新记录。处理页处理的程序代码如下：

例程12　代码位置：光盘\TM\08\online\admin\dataadd_chk.php



```
<?php
    session_start();                                          //开启 session 支持
    include "inc/chec.php";                                   //载入权限检查文件
    include "conn/conn.php";                                  //载入数据库链接文件
    include "inc/func.php";                                   //载入自定义函数文件
    $p_type = array("jpg","jpeg","bmp","gif");                //允许上传的图片文件格式
    $f_type = array("avi","rm","rmvb","wav","mp3","mpg");     //允许上传的视频、音频格式
    $audio_path = "../upfiles\\audio";                        //视频上传位置
    $video_path = "../upfiles\\video";                        //音频上传位置
    $picture_path ="";                                        //图片上传路径
    $file_path = "";                                          //文件上传路径
```

在图片类型判断中，首先判断图片大小是否在允许范围之内，接着使用自定义函数 f_postfix()来判断图片后缀是否符合要求，如果没有问题，则根据 time()函数生成图片名，并保存到变量$picture_path 中。

例程13　代码位置：光盘\TM\08\online\admin\dataadd_chk.php

```
        /*判断图片大小，是否大于 0KB，小于 700KB*/
❶  if($_FILES[picture][size] > 0 and $_FILES[picture][size] < 700000){
        /*如果图片格式正确*/
❷      if(($postf = f_postfix($p_type,$_FILES[picture][name])) != false){
        /*使用 time()函数生成文件名*/
❸          $picture_path = time().".".$postf;
        /*如果 action 值为 a 时，说明上传为视频图片*/
            if($_POST[action] == "a"){
                /*如果生成的临时文件正确，则使用 move_uploaded_file() 函数上传*/
                if($_FILES[picture][tmp_name])
❹                  move_uploaded_file($_FILES[picture][tmp_name],$audio_path."\\".$picture_path);
                else{
                    echo "<script>alert('上传图片失败！');history.go(-1);</script>";
                    exit();
                }
                /*如果上传图片为音频图片*/
            }else if($_POST[action] == "v"){
                if($_FILES[picture][tmp_name])
                    move_uploaded_file($_FILES[picture][tmp_name],$video_path."\\".$picture_path);
                else{
                    echo "<script>alert('上传图片失败！');history.go(-1);</script>";
                    exit();
                }
            }
        }else{
            echo "<script>alert('上传图片格式错误！');history.go(-1);</script>";
            exit();
        }
                /*如果图片大于 700KB，说明图片太大，超出要求，返回上一步*/
    }else if($_FILES[picture][size] > 700000){
            echo "<script>alert('上传图片大小超出范围！');history.go(-1);</script>";
            exit();
    }
    else{
        $picture = "";
```

```
    }
    /*******************************/
```

代码贴士

❶ 使用预定义变量的$_FILES[name][size]属性判断上传文件的大小。如果文件大于 pho.ini 中设置的上传文件的最大值，那么 size 将永远返回 0，而不是文件的实际大小。

❷ 使用自定义函数 f_postfix()判断上传文件的后缀。

❸ 返回当前 UNIX 的时间戳。

❹ 使用 move_uploaded_file()上传函数要注意：它的第一个参数是上传到服务器中的临时文件名，而不是文件原始名称，是用户自定义的名称。要取得临时文件名使用$_FILES[name][tmp_name]。

对上传文件的判断和上传图片的流程基本相同，也是先判断文件大小是否符合上传文件的范围要求，接着判断上传文件后缀，最后使用 time()函数生成文件名，保存到变量$address_path 中。下面是判断视频文件的流程，代码如下：

例程14 代码位置：光盘\TM\08\online\admin\dataadd_chk.php

```
    /*判断上传文件类型与大小*/
    if($_FILES[address][size] > 0){                              //判断上传文件大小
        if($_POST[action] == "a"){                               //判断是视频文件还是音频文件
            if($_FILES[address][size] < 300000000){              //判断文件是否超出限定大小
                if(($postf = f_postfix($f_type,$_FILES[address][name])) != false){
                    $file_path = time().".".$postf;              //生成新的文件名
                    if($_FILES[address][tmp_name])               //如果临时文件正确
    /*使用 move_uploaded_file() 函数上传文件*/
                        move_uploaded_file($_FILES[address][tmp_name],$audio_path."\\".$file_path);
                    else{
                        echo "<script>alert('上传文件错误！');history.go(-1);</script>";
                        exit();
                    }
                }else{
                    echo "<script>alert('上传文件格式错误！');history.back(-1);</script>";
                    exit();
                }}else{
                echo "<script>alert('上传文件大小错误！');history.go(-1);</script>";
                exit();
            }}
        /*如果是音频文件*/
        else if($_POST[action] == "v"){
        /*判断文件大小*/
            if($_FILES[address][size] < 10000000){
        /*判断文件后缀*/
                if(($postf = f_postfix($f_type,$_FILES[address][name])) != false){
                    $file_path = time().".".$postf;
        /*判断临时文件是否生成*/
                        if($_FILES[address][tmp_name])
        /*使用 move_uploaded_file() 函数上传文件*/
                        move_uploaded_file($_FILES[address][tmp_name],$video_path."\\".$file_path);
                    else{
```

```
                    echo "<script>alert('上传文件错误！ ');history.go(-1);</script>";
                    exit();}}else{
                echo "<script>alert('上传文件格式错误！ ');history.back(-1);</script>";
                exit();}}else{
            echo "<script>alert('上传文件大小错误！ ');history.go(-1);</script>";
            exit();}}
    }else{
        echo "<script>alert('没有上传文件或文件大于 300MB');history.go(-1);</script>";
        exit();
    }
```

如果上传图片和上传的文件都没有问题，则对表单的其他信息进行处理，即生成 insert 插入语句，添加新记录。程序代码如下：

例程15　代码位置：光盘\TM\08\online\admin\dataadd_chk.php

```
    /*视频文件和音频文件相同的信息*/
    $names = $_POST[names];                          //视频名称
    $grade = $_POST[grade];                          //级别
    $sizes = $_FILES[address][size];                 //大小
    $publisher = $_POST[publisher];                  //发行商
    $actor = $_POST[actor];                          //演员
    $language = $_POST[language];                    //语言
    $style = $_POST[style];                          //类别
    $types = $_POST[types];                          //类型
    $froms = $_POST[from];                           //发行国家
    $publishtime = $_POST[publishtime];              //首映时间
    $news = $_POST[news];                            //是否为新片
    $remark = $_POST[remark];                        //备注

    /****************/
    if($_POST[action] == "a"){
        /*确认父级目录*/
        /*表单值，视频文件特有的属性*/
        $director = $_POST[director];                //提取导演
        $marker = $_POST[marker];                    //制片人
        /*生成 insert 语句*/
        $a_sqlstr = "insert into tb_audio (name,picture,sizes,grade,publisher,actor,director,marker,languages,type,
style,froms,publishtime,bool,remark, property,address,username,issueDate) values('$names','$picture_path','$sizes',
'$grade','$publisher','$actor', '$director', '$marker','$language','$types', '$style','$from', '$publishtime','$news','remark', '
管理员','$file_path','$_SESSION[admin]','".date("Y-m-d H:i:s")."')";
    }
    else if($_POST[action] == "v"){                  //如果上传文件为音频文件
        $actortype = $_POST[actortype];              //歌曲类型
        $ci = $_POST[ci];                            //作词
        $qu = $_POST[qu];                            //作曲
        /*生成 insert 语句*/
        $a_sqlstr = "insert into tb_video (name,picture,actor,ci,qu,actortype,type,style,publisher,froms,sizes,
```

```
languages,publishTime,remark,property,address,userName,issueDate) values('$names','$picture_path', '$actor',
'$ci','$qu','$actortype','$types','$style','$publisher','$froms','$sizes','$language','$publishtime','$remark',' 管 理 员 ','$file_
path','$_SESSION[admin]','".date("Y-m-d H:i:s")."')";
    }else{
        echo "<script>alert('错误');window.close();</script>";
        exit();
    }
    /*上传图片和文件*/
    /***************************/
    $a_rst = $conn->execute($a_sqlstr);
    if(!($a_rst == false))
        echo "<script>top.opener.location.reload();alert('添加成功');window.close();</script>";
    else
        echo "<script>alert('添加失败');history.go(-1);</script>";
?>
```

8.11.4　数据删除的实现过程

数据删除使用的数据表：tb_audio、tb_video

视频数据删除主要是根据传递的文件 id 值，将数据库中的记录删除，同时使用 unlink()函数删除服务器端的真实数据。程序代码如下：

例程 16　代码位置：光盘\TM\08\online\admin\del_video_chk.php

```
<?php
    session_start();                                        //开启 session 支持
    include "conn/conn.php";                                //载入数据库链接文件
    include "inc/chec.php";                                 //载入权限判断文件
    $file_path = "../upfiles/video/";                       //生成文件路径
    $s_sqlstr = "select * from tb_video where id = ".$_GET[id];   //生成 SQL 语句
    $s_rst = $conn->execute($s_sqlstr);
    if(!($s_rst == false)){
        /*删除相应的图片文件和音频、视频文件*/
        if(unlink($file_path.$s_rst->fields[16]) and unlink($file_path.$s_rst->fields[2])){
        /*生成删除语句*/
            $d_sqlstr = "delete from tb_video where id = ".$_GET[id];
        /*执行删除语句*/
            $d_rst = $conn->execute($d_sqlstr);
            if(!($d_rst == false)){
                echo "<script>alert('删除成功');location='main.php?action=video';</script>";
            }else{
                echo "<script>alert('删除失败');history.go(-1);</script>";
            }}else
                echo "<script>alert('删除失败');history.go(-1);</script>";
?>
```

8.12　管理员设置模块设计

8.12.1　管理员设置模块概述

管理员设置模块主要包括管理员信息添加、管理员信息删除及管理员信息冻结 3 个部分。管理员可以通过后台管理功能导航进入管理员设置页面（member.php ）。在管理员设置页面中，管理员可以进行管理员信息的添加操作，或进行会员信息的冻结或删除等操作。管理员设置用例图如图 8.35 所示。

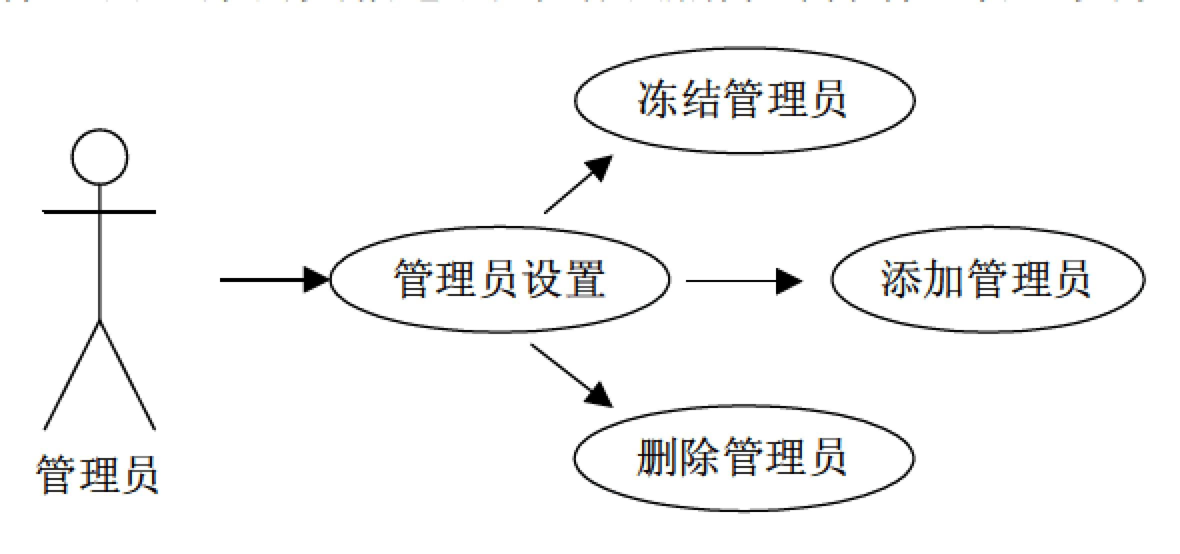


图 8.35　管理员设置用例图

8.12.2　管理员设置模块技术分析

管理员可以单击管理员设置页面的“冻结”或“解冻”按钮，对管理员信息进行冻结或解冻的操作。当管理员登录时，首先查看数据库中的 whether 字段，如果为 1，则说明该用户处于激活状态；否则，将显示提示窗口，告诉该用户此账号已被冻结等提示信息。

对管理员信息的冻结或解冻操作主要是根据传递的管理员的 id，将数据库中的 whether 字段更新。程序代码如下：

例程 17　代码位置：光盘\TM\08\online\admin\m_freeze_chk.php

```
<?php
	session_start();								//开启 session 支持
	include "conn/conn.php";						//引入数据库链接文件
	include "inc/chec.php";							//引入权限检查文件
	/*根据 whether 值，进行取反操作*/
	if($_POST[whether] == "1")
		$wt = "0";
	else if($_POST[whether] == "0")
		$wt = "1";
	else{
		echo "<script>alert('非法操作!');history.go(-1);</script>";
		exit();
	}
	/*根据管理员 id 和赋值后的$wt，生成 update 语句*/
	$o_sqlstr = "update tb_manager set whether = '".$wt."' where id = ".$_POST[id];
	/*执行 update 语句*/
	$o_rst = $conn->execute($o_sqlstr);
```

```
        if(!($o_rst == false)){
            echo "<script>alert('操作成功');location='main.php?action=manager';</script>";
        }
?>
```

该方法实现很简单，但却十分实用，在实行会员制的网站上，在可以发帖、回复的论坛中，都可以使用这个方法来管理。

8.12.3 管理员添加的实现过程

管理员添加使用的数据表：tb_manager

管理员可以通过管理员设置页面打开管理员信息添加页面（addmanager.php）。在管理员信息添加页面中，管理员可以填写表单并将表单提交到处理页进行数据处理，以完成管理员信息添加的操作。管理员添加页面的运行结果如图 8.36 所示。



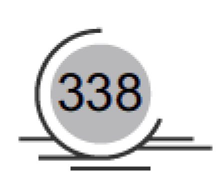
图 8.36 管理员添加页面的运行结果

管理员添加页面涉及的 HTML 表单元素如表 8.5 所示。

表 8.5 管理员添加页面涉及的 HTML 表单元素

名 称	类 型	重 要 属 性	含 义
list	form	method="post" action="addmanager_chk.php"	表单
names	text	id="names" size="30"	管理员名称
password	password	id="password" size="30"	密码
password2	password	id="password2" size="30"	密码确认
grade	select	<option value="视频目录管理员" selected>视频目录管理员</option> <option value="音频目录管理员">音频目录管理员</option> <option value="视频数据管理员">视频数据管理员</option> <option value="音频数据管理员">音频数据管理员</option> <option value="会员数据管理员">会员数据管理员</option> <option value="会员等级管理员">会员等级管理员</option>	管理权限
realname	text	id="realname" size="30"	真实姓名
Submit	button	class="submit" onClick="check();"	“添加”按钮
Submit2	button	value="返　回" class="submit" onClick="javascript:top.window.close()"	“返回”按钮

管理员信息添加的数据处理主要是根据提交的表单数据，在数据库中添加对应的记录，首先判断



输入的名称是否存在，如果存在，显示提示语，并返回到上一步；如果不存在这种错误，生成 insert 语句，添加新管理员。程序代码如下：

例程18　代码位置：光盘\TM\08\online\admin\addmanager.php

```
<?php
    session_start();                                                    //开启 session 支持
    include "conn/conn.php";                                            //载入数据库链接文件
    include "inc/chec.php";                                             //载入权限检查文件
    $a_sql="select * from tb_manager where name='".$_POST[names]."'";   //检查是否有重名
    $a_rst = $conn->execute($a_sql);
    /*如果管理员名称存在，弹出提示框，并返回到上一步*/
    if(!$a_rst->EOF)
        echo "<script>alert('该名称的管理员已经存在，请更换名称');history.go(-1);</script>";
    else{
    /*生成 insert 语句*/
        $a_sqlstr="insert into tb_manager values('','".$_POST[names]."','".$_POST[password]."','".$_POST
[grade]."','".$_POST[realname]."','".date("Y-m-d"). "','1')";
    /*执行 insert 语句*/
        $a_rst1 = $conn->execute($a_sqlstr);
    /*如果管理员添加成功*/
        if(!($a_rst1 == false)){
?>
        <script>
    /*刷新父窗口，弹出提示框，最后删除当前窗口*/
            top.opener.location.reload();
            alert("管理员添加成功");
            top.window.close();
        </script>
<?php
        }
        else
            echo "<script>alert('添加失败".$a_sqlstr."');history.go(-1);</script>";
    }
?>
```

8.12.4　管理员删除的实现过程

管理员删除使用的数据表：tb_manager

管理员信息删除主要是根据传递的管理员的 id，将数据库中对应的数据删除。程序代码如下：

例程19　代码位置：光盘\TM\08\online\admin\ del_mfreeze_chk.php

```
<?php
    session_start();                                            //开启 session 支持
    include "conn/conn.php";                                    //载入数据库链接文件
    include "inc/chec.php";                                     //载入权限检查文件
    $d_sqlstr = "delete from tb_manager where id = ".$_GET[id]; //生成 delete 语句
    if(!($d_rst = $conn->execute($d_sqlstr)) == false)          //判断删除是否成功
        echo "<script>alert('删除成功');location='main.php?action=manager';</script>";
    else
```

```
        echo "<script>alert('删除失败');history.go(-1);</script>";
?>
```

8.13 开发技巧与难点分析

在本系统中，多个地方使用了无边框窗口和刷新父级窗口的技术。下面来学习一下有关这两方面的知识要点。

8.13.1 无边框窗口

平时看到的浏览器窗口都是有边框的，如果偶尔有一个无边框的窗口弹出，那么将会更加引人注目。实现无边框窗口使用的是 JavaScript 脚本中的 open()方法，该方法的常用格式如下：

```
window.open(url,name,features,replace);
```

open()方法中的参数说明如表 8.6 所示。

表 8.6 open()方法中的参数说明

属 性 值	说 明
url	在弹出窗口中要打开的地址
name	要打开窗口的名字。在使用 target 属性时会用到，可以为空
features	列举的窗口特征。如果不写任何参数，那么默认的参数多数情况下是关闭的
replace	一个 boolean 值，指出是否使用 url 替换当前内容。该参数可以省略不写

下面应用 window 对象的 open()方法打开一个指定大小的窗口。代码如下：

```
<script language="javascript">
    wopen = open("list.php","query","height=500,width=665,scrollbars=no,toolbar=yes,location=yes");
</script>
```

运行结果如图 8.37 所示。



图 8.37 使用 open()方法弹出窗口

除了上面实例中给出的几个窗口特征，表 8.7 还列出了其他几个常用的特征参数及说明。



表 8.7　浏览器窗口的外观样式

属　性　值	说　　明
left	新窗口的左坐标
top	新窗口的上坐标，配合 left 可以定位一个窗口的弹出位置
resizable	是否可以通过拖曳来调整新窗口的大小，默认为 no
status	在新窗口中是否显示状态栏，默认为 no

注意　窗口特征是用等号赋值、逗号分割的，在逗号或等号前后不要加上空格，因为在有些浏览器中会显示错误。

8.13.2　刷新父级窗口和关闭子窗口

在打开的子窗口进行一系列的操作，如添加、删除数据后，就要对当前窗口进行关闭，最后返回父级窗口中，但父级窗口中的数据可能仍然是没有添加或删除之前的记录数，这时就可以调用 JavaScript 脚本对父窗口进行刷新了。这种对父级窗口的刷新只有对弹出窗口才有效，使用时要注意。该脚本的使用格式如下：

```
top.opener.location.reload();
```

operner 属性是子窗口的最高层 window 对象才有的属性，配合 location 对象可以对父窗口进行刷新或重载。location 对象的两个常用方法有 href 方法和 reload 方法。href 方法可以刷新父窗口内容，使用格式如下：

```
top.opener.location.href="operation.php";
```

关闭子窗口使用如下代码即可：

```
window.close();
```

不过这里要注意的是：close()方法在关闭子窗口时，不弹出提示就直接关掉浏览器了，但对于父级窗口，则弹出提示框，经确认后才可以关闭浏览器。

8.14　ADODB 连接数据库技术专题

PHP 为各种版本的数据库定义了庞大的函数群来支持，而且支持得很完整，那为什么本系统要使用 ADODB 来对数据库进行操作呢？原因在于 ADODB 的优势：一次编写，不需修改。也就是说，当系统使用的数据库需要更换时，只要更新数据库、数据表即可，不需要再重新编写源码了，这为后期维护节约了一大笔开销，而且不需要长时间关闭网站，这是 PHP 函数所无法比拟的。本章的技术专题就来着重学习一下 ADODB 技术。

8.14.1 ADODB 简介

由于 PHP 存取函数没有标准化，不同数据库间的函数名称、参数差异很大，在更换数据库时，会带来大量的代码修复工作。这时就需要一组函数库来隐藏不同资料库函数界面间的差异，让开发者可以很简单地去切换资料库，这就是 ADODB。ADODB 目前支持 MySQL、Oracle、Microsoft SQL Server、Sybase、PostgreSQL、Foxpro、Access、ADO 和 ODBC 等大多数知名的、不知名的数据库类型。

8.14.2 ADODB 的安装

要使用 ADODB 来操作数据库，首先就要获取和安装 ADODB。读者只要到网上下载 ADODB 包，解压到 Web 服务器目录下即可。


注意 要使用 ADODB，使用的 PHP 必须是 4.01 以上的版本。

8.14.3 ADODB 常用函数及说明

在 8.5.2 节中已经介绍过了使用 ADODB 的一般步骤，在实际使用中，需要根据情况进行不同的处理，这时就需要对步骤及功能进行扩展，下面就来介绍几种经常会用到的函数和常量。

1. PConnect($host,$user,$password, $database)

参数说明如下。

☑ $host：数据库服务器地址。
☑ $user：数据库用户名。
☑ $password：数据库密码。
☑ $database：连接的数据库。

如果连接成功回传 true，失败则回传 false。

2. SelectLimit($sql,$numrows=-1,$offset=-1)

参数说明如下。

☑ $sql：执行的 SQL 语句。
☑ $numrows：显示的记录条数。
☑ $offset：从第几条记录开始计算。

该函数和 MySQL 中的 limit 子句相似，但要注意，limit 后面的两个参数的顺序是（limit offset,numrows），该函数是完全相反的。

3. ErrorMsg()

回传错误信息。当 Execute()函数出现错误时，使用 ErrorMsg()函数可以查看最后出错的相关信息。使用方法如下：

```
…
$recordset = $conn->Execute("select * from tb_audio")
```



```
if(!$recordset)                                                //判断 execute()函数执行中是否有错误
    echo $conn->ErrorMsg();
else{
    …
}
```

即使没有出现错误，ErrorMsg()也会回传最后的状态值，这种情况下一般不需要输出。

4．RecordCount()

回传结果集中的记录数。使用方法如下：

```
…
$sql = 'select * from tb_account';
$rs = $conn->Execute($sql);                                    //执行 SQL 语句
echo $rs->RecordCount();                                       //输出查询结果数
…
```

5．FieldCount()

回传结果集中的字段数。使用方法同上面的 RecordCount()函数。

6．MoveNext()、Move()、MoveFirst()、MoveLast()

- ☑ MoveNext()：移动指针到一下条记录。
- ☑ Move($to)：移动指针到指定的列（$to）。这里使用的是绝对定位，如果$to 是 10，就移动到结果集中的第 10 条记录；如果$to 大于记录总数，指针将移动到最后。需要注意的是，部分数据库不支持向后移动指针。
- ☑ MoveFirst()：实际上就是 Move(0)，指针移动到结果集头部，部分数据库不支持该函数。
- ☑ MoveLast()：指针移动到结果集的尾部，也可以表示为 Move($recordCount() - 1)，部分数据库不支持此函数。

7．$ADODB_FETCH_MODE

该常量决定了结果集以哪种方式将数据回传。在有些数据库中，使用$recordset->fields[]仅支持字段编号为索引，即 0、1、2 等，不支持字段名（如 id，user）作索引，这时可以使用该常量进行设置，该常量的值有：

```
define('ADODB_FETCH_DEFAULT',0);                    //系统默认值
define('ADODB_FETCH_NUM',1);                        //以字段编号为索引
define('ADODB_FETCH_ASSOC',2);                      //以字段名称为索引
define('ADODB_FETCH_BOTH',3);                       //同时用时编号和名称
```

这时上面的步骤可以改为：

```
…
$conn = &ADONewConnection("mysq");
$conn->PConnect("db_online");
$ADODB_FETCH_MODE = ADODB_FETCH_ASSOC;
$recordset = $conn->Execute("select * from tb_audio");
```

```
if(!$recordset)                                    //判断 Execute()函数执行中是否有错误
      echo $conn->ErrorMsg();
else{
      while(!$recordset->EOF){                     //如果没有错误，则配合 wihle 语句循环输出结果
            echo $recordset->fields[id]."<br>";
            $recordset->movenext();                //指针下移
      }
}
…
```

这里建议将$ADODB_FIELDS_MODED 的值固定为 ADODB_FETCH_NUM 及 ADODB_FETCH_ASSOC，不建议使用 ADODB_FETCH_BOTH，因为许多驱动程序并不支持该常量。

ADODB 的常用函数就介绍这么多了，如果读者感兴趣，可以下载 ADODB 手册来学习更多内容。

8.14.4 不同数据库之间的转换

使用 ADODB 的优势就是可以在以后的升级中，更换不同的数据库而不用大量地改动源程序。但不同数据库之间，有的 SQL 语句或语法是不相同的，这时就要做一些小幅的调整。

（1）在 conn.php 文件中，更改连接的数据库类型和参数。

```
$conn = &ADONewConnection('mysql');                        //建立 MySQL 连接
$conn->PConnect("localhost","root","root","db_online");   //连接 db_online 数据库
```

如使用 SQL Server 时，上面的链接语句更改为：

```
$conn = &ADONewConnection('mssql');
$conn->PConnect("localhost","sa","","db_online");
```

（2）注意包含时间类型的 SQL 语句。

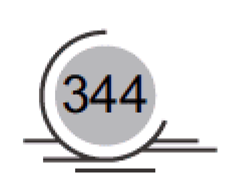
不同的数据库对于时间的表示方法是不同的，如 MySQL 使用时间类型时，就像使用字符类型一样，放到单引号中就可以了，但 Access 则需要放到“#”之间。Oracle 虽然也是使用单引号，但它的格式为“日-月-年”，而其他数据库大多都是“年-月-日”。

（3）尽量使用通用的 SQL 语句。

尽管国际标准化组织（ISO）在 1987 年创建了 SQL 的一个标准，但在各种 SQL 之间仍然存在差别。所以为了增加通用性，最好了解一下数据库间的不同。例如，MySQL 中的 limit 子句在 Oracle 中就不被支持。

8.15 本章总结

online 影视 365 网实现了一个在线播放平台。该网站具备一个影视网站所需要的所有的基本功能，包括影视文件的上传、下载、播放及介绍。还介绍了 JavaScript 脚本语言的一些实用技巧，最后介绍了本章的核心内容：ADODB 连接数据库技术。希望通过本章的学习，读者不仅可以建立一个自己的影视网，而且可以熟练使用 ADODB 来连接任意一个数据库。有条件的读者可以试着将本系统的数据库改为 SQL Server 或 Oracle，来增加对 ADODB 的了解及熟练程度。



第 9 章

明日知道网站

（Zend Framework 实现）

在全球信息化高速发展的今天，互联网已经成为资源共享、互动、交流、办公及娱乐的最为广泛和有效的平台，如何高效地利用这些资源已经成为网民们最为关注的问题。百度、谷歌等互联网搜索工具的出现已经在很大程度上提高了信息共享的进程。本章所介绍的明日知道网站是针对 IT 技术问题进行搜索、为程序开发人员提供交互的平台，与互联网上一些知名的搜索工具相比更具专业化特点。

通过阅读本章，可以学习到：

- 网站项目开发的基本流程
- Zend Framework 框架的搭建方法
- MVC 开发模式的应用
- 类似 Google 搜索条的制作方法
- 查询结果的关键字描红输出
- 在线编辑器的原理及制作方法

9.1 开发背景

××科技有限公司是一家以 IT 教育、互联网开发和运营为主的高科技互联网企业。在多年的经营中积累了大量的 IT 技术方面的资源，为了提升公司的知名度，使资源得以共享，并创造更大的广告利润，公司组织了一支团队来开发明日知道网站，争取打造一个国内知名的专业 IT 技术搜索和交互平台。

明日知道网站是为了使搜索变得更加专业，以方便广大程序相关工作者而打造的专业化互联网搜索引擎。通过该搜索引擎，程序开发人员可以方便地查找到有关软件开发方面的技术问题和文章。并且明日知道网站提供了互动平台，在学习编程过程中如果遇到了技术问题，可以将问题发表于明日知道网站上，其他注册会员可以对提出的问题进行解答，这样即便通过明日知道没有搜索到要查找的内容，也可以通过互动方式将问题解决。专业化搜索工具是有一定发展潜能的项目课题。

9.2 需求分析

对于明日知道这类信息检索类网站来说，功能强大的搜索模块显得尤为重要，借鉴其他知名的搜索工具，方便、高效的搜索引擎应该具备关键字提示列表、分词查询和查询结果的描红输出等功能。为了避免用户要查找的内容在本站没有查找到结果，所以还需要增加用户交流模块，这样即使用户的问题在本站没有查找到答案，也可以将问题发布出去，其他用户就可以对该问题进行回答，直到解决为止。所以明日知道网站应该打造一个服务开发人员，对资源进行不断积累的综合平台。

通过实际调查，要求明日知道网站应该具备以下功能：

☑ 类似百度和 Google 等搜索引擎的搜索条。
☑ 多关键字的分词查询。
☑ 查询结果的关键字描红输出。
☑ 使用在线编辑器发布和回复问题。
☑ 问题的引用。

9.3 功能结构

“磨刀不误砍柴工”，建立一个完善的系统功能图对项目的整体开发具有指导意义，因此在项目编码前应给出项目前台和后台的系统功能结构图。其中，明日知道网站前台功能结构图如图 9.1 所示，后台功能结构图如图 9.2 所示。

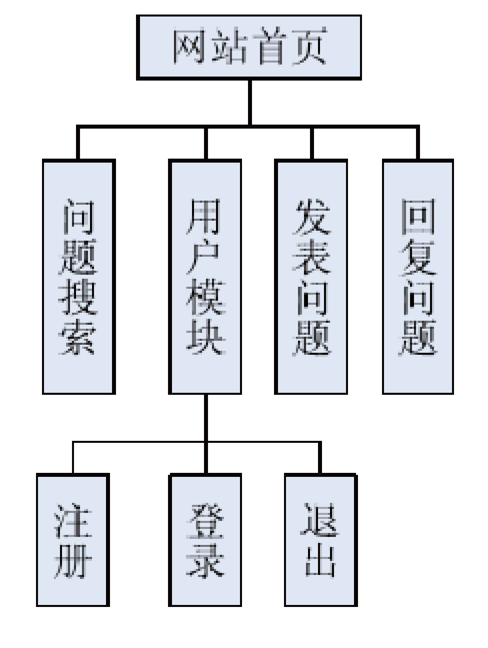


图 9.1　网站前台功能结构图

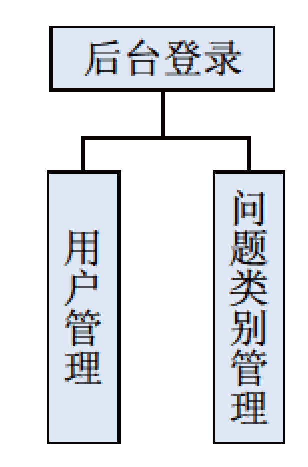


图 9.2　网站后台功能结构图

9.4　功 能 预 览

明日知道网站由多个功能模块组成，为了让读者对本系统有个初步的了解和认识，下面列出几个典型功能的页面，其他页面参见光盘中的源程序。

明日知道首页由网站 Logo、语言导航条、搜索区、友情链接和版权声明等部分组成。页面简洁大方、突出核心，便于用户使用，其运行效果如图 9.3 所示。

图 9.3　明日知道首页

会员登录模块完成注册会员的登录操作，并且融入验证码技术，同时加入了会员注册的超链接，其运行效果如图 9.4 所示。

问题分类模块展示明日知道网站中专题技术的分类，同时展示每类中包含的问题和回复情况，其运行效果如图 9.5 所示。



图 9.4　会员登录页面



图 9.5　问题分类页面

输出查询结果模块展示搜查的结果，并且对查询的关键字进行描红，同时对查询结果进行分页输出，其运行效果如图 9.6 所示。



图 9.6　查询结果的分页输出页面

9.5 数据库设计

在进行项目开发前需要根据实体属性建立数据库实体模型，然后根据实体之间存在的客观联系设计出合理的数据表。由于数据库是项目内容的主要载体，所以设计出合理的数据库、表及表间关系显得非常重要。

9.5.1 数据库分析

明日知道网站采用 PHP 语言开发，并且该项目为资源搜索型网站，数据量会很大，所以数据库稳定、高效的存储和检索能力显得非常重要，综合上述因素及 PHP 的完全跨平台等特性，应该首选 MySQL 数据库作为明日知道网站数据的载体。在长时间的工作中也总结出，PHP 和 MySQL 是最佳的黄金搭档。

9.5.2 数据库概念设计

根据明日知道网站的需求分析、功能结构分析，规划出整个项目的数据库实体应该包括问题类别实体、关键字实体、问题实体、回复实体和用户实体等部分。下面介绍这几个实体的 E-R 图。

1．问题类别实体

明日知道网站定位打造一个专业、高效的在线搜索工具，这样就需要搜索内容按类别划分。资源分类明确，可以有效地提高用户检索的效率，方便用户定位查找。问题类别实体包括类别 ID、类别名称、类别描述和添加时间等属性，其实体 E-R 图如图 9.7 所示。

2．问题实体

问题实体是对用户提出问题的抽象，根据功能分析，问题实体包括问题 ID 号、主题、内容、添加时间、置顶标识、用户 ID、类别 ID、浏览次数、上传文件名和精华问题标识等属性，其实体 E-R 图如图 9.8 所示。

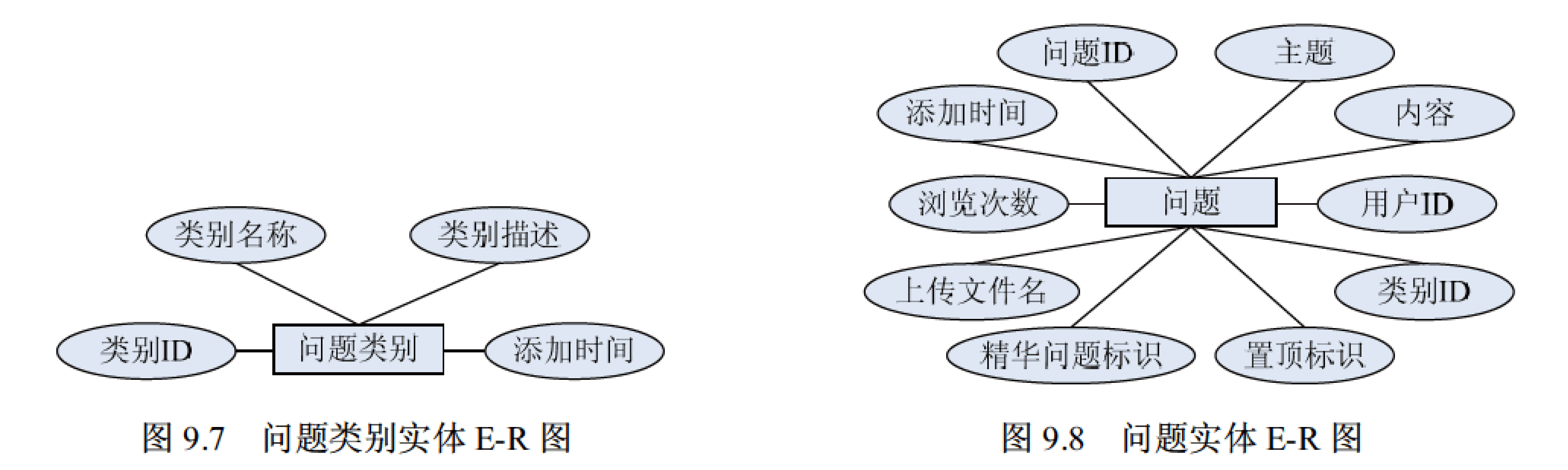


图 9.7 问题类别实体 E-R 图

图 9.8 问题实体 E-R 图

3．回复实体

回复实体包括回复 ID 号、主题、内容、最佳答案标识、添加时间、用户 ID、主题 ID 和类别 ID 等属性，其实体 E-R 图如图 9.9 所示。



4．关键字实体

关键字实体是对用户输入关键字进行记录的抽象，该实体包括 ID 号、关键字和检索次数等属性，其实体 E-R 图如图 9.10 所示。

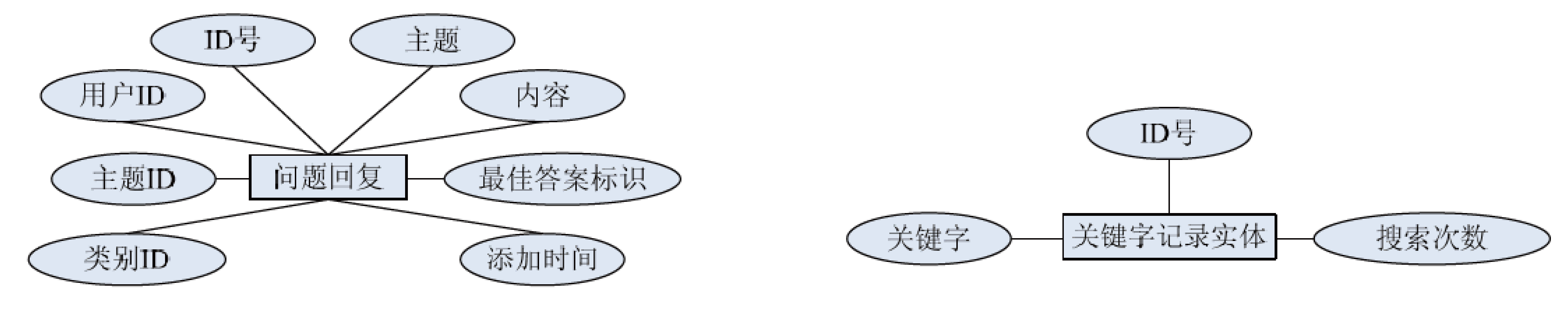


图 9.9　回复实体 E-R 图　　　图 9.10　关键字实体 E-R 图

5．用户实体

用户实体是对网站注册用户的抽象，其属性包括用户 ID、用户昵称、登录密码、E-mail、联系电话、所在地、注册时间、头像名称、发帖次数、回帖次数和用户类别标识等属性，其实体 E-R 图如图 9.11 所示。

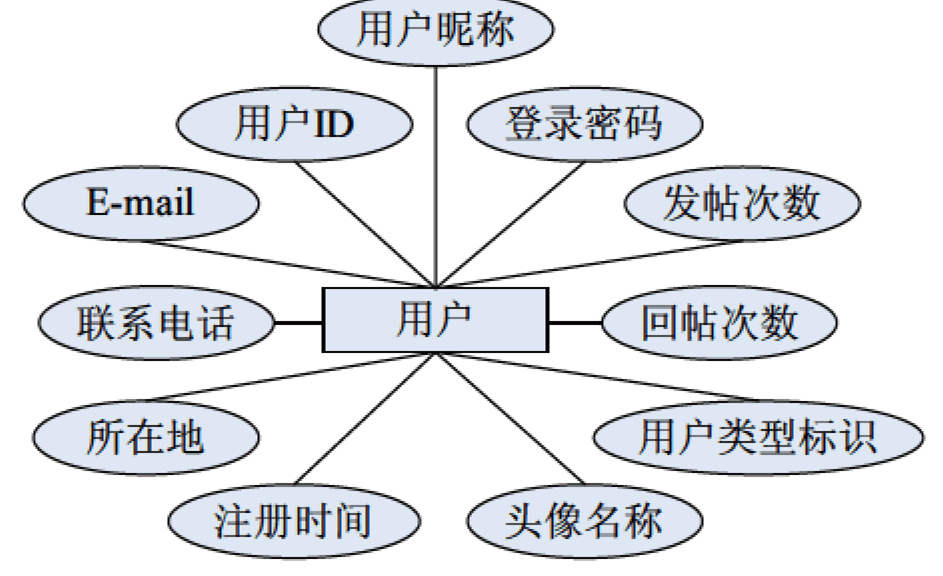


图 9.11　用户实体 E-R 图

9.5.3　创建数据库及数据表

根据需求分析及数据库概念设计，可以制定出明日知道网站数据库（这里命名为 db_known）应该包括问题类别表（tb_bbstype）、问题表（tb_title）、回复表（tb_reply）、关键字表（tb_keywords）和用户表（tb_user）等 5 个表，其中各表的主要参数如图 9.12 所示。

下面将具体介绍明日知道数据库中各个数据表字段的具体描述。

1．问题类别表（tb_bbstype）

问题类别表（这里命名为 tb_bbstype）用于存储问题类别，其结构如图 9.13 所示。

表	类型	整理	大小
tb_bbstype	MyISAM	utf8_unicode_ci	2.6 KB
tb_keywords	MyISAM	utf8_unicode_ci	4.5 KB
tb_reply	MyISAM	utf8_unicode_ci	4.7 KB
tb_title	MyISAM	utf8_unicode_ci	6.6 KB
tb_user	MyISAM	utf8_unicode_ci	10.8 KB

图 9.12　明日知道网站数据表主要参数

字段	类型	整理	属性	Null	默认	额外
id	int(11)			否		auto_increment
typename	varchar(50)	utf8_unicode_ci		否		
addtime	datetime			否		
description	varchar(255)	utf8_unicode_ci		否		

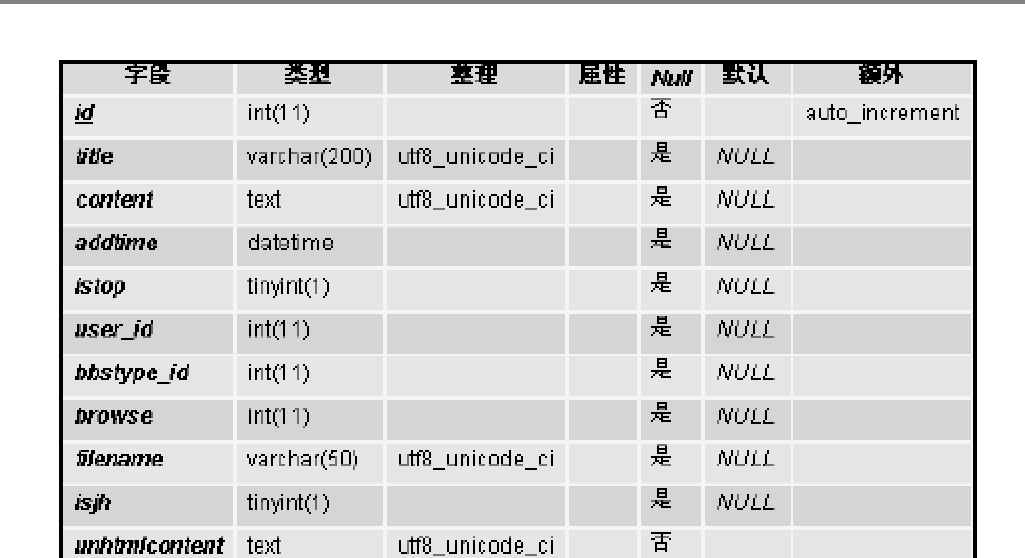
图 9.13　问题类别表结构

2．问题表（tb_title）

问题表（这里命名为 tb_title）用于存储问题内容，其结构如图 9.14 所示。

3．回复表（tb_reply）

回复表（这里命名为 tb_reply）用于存储用户对问题的回复，其结构如图 9.15 所示。



字段	类型	整理	属性	Null	默认	额外
id	int(11)			否		auto_increment
title	varchar(200)	utf8_unicode_ci		是	NULL	
content	text	utf8_unicode_ci		是	NULL	
addtime	datetime			是	NULL	
istop	tinyint(1)			是	NULL	
user_id	int(11)			是	NULL	
bbstype_id	int(11)			是	NULL	
browse	int(11)			是	NULL	
filename	varchar(50)	utf8_unicode_ci		是	NULL	
isjh	tinyint(1)			是	NULL	
unhtmlcontent	text	utf8_unicode_ci		否		

图 9.14　问题表结构

字段	类型	整理	属性	Null	默认	额外
id	int(11)			否		auto_increment
title	varchar(200)	utf8_unicode_ci		是	NULL	
content	text	utf8_unicode_ci		是	NULL	
addtime	datetime			是	NULL	
user_id	int(11)			是	NULL	
title_id	int(11)			是	NULL	
topindex	int(11)			是	NULL	
bbstype_id	int(11)			是	NULL	

图 9.15　回复表结构

4．关键字表（tb_keywords）

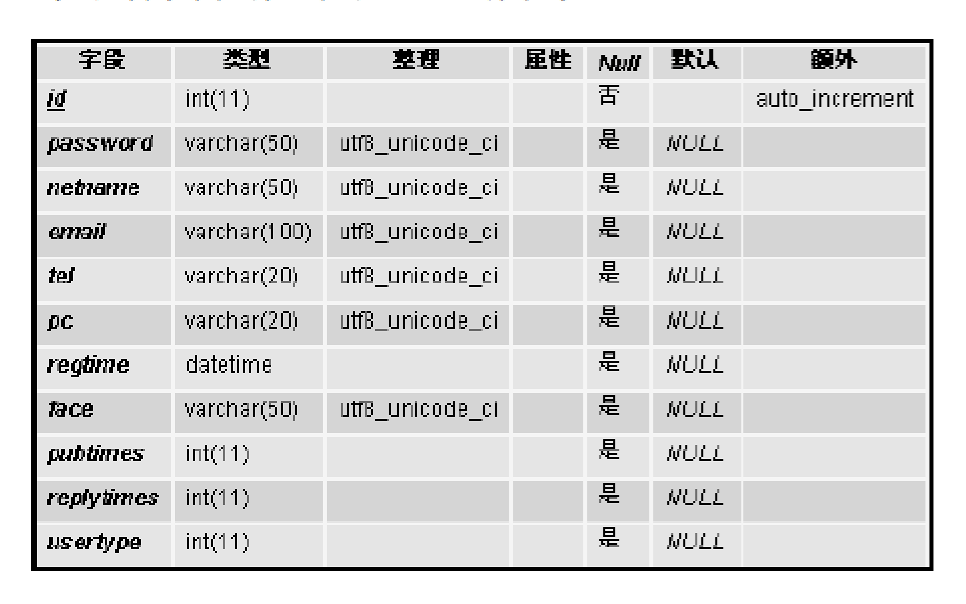
关键字表（这里命名为 tb_keywords）用于存储用户曾搜索的关键字及每个关键字被搜索的次数，其结构图如图 9.16 所示。

5．用户表（tb_user）

用户表（这里命名为 tb_user）用于存储用户信息，其结构图如图 9.17 所示。

字段	类型	整理	属性	Null	默认	额外
id	int(11)			否		auto_increment
keyword	varchar(50)	utf8_unicode_ci		否		
searchtime	int(11)			否		

图 9.16　关键字表结构



字段	类型	整理	属性	Null	默认	额外
id	int(11)			否		auto_increment
password	varchar(50)	utf8_unicode_ci		是	NULL	
netname	varchar(50)	utf8_unicode_ci		是	NULL	
email	varchar(100)	utf8_unicode_ci		是	NULL	
tel	varchar(20)	utf8_unicode_ci		是	NULL	
pc	varchar(20)	utf8_unicode_ci		是	NULL	
regtime	datetime			是	NULL	
face	varchar(50)	utf8_unicode_ci		是	NULL	
pubtimes	int(11)			是	NULL	
replytimes	int(11)			是	NULL	
usertype	int(11)			是	NULL	

图 9.17　用户表结构

9.6　技 术 攻 关

明日知道网站在开发过程中，多处使用了较新的技术。为了便于对该项目各个模块的理解，本节首先对明日知道网站所使用的关键技术进行讲解。

9.6.1　类似 Google 搜索引擎的搜索条

为了方便网站浏览者检索网站内容，很多知名网站的查询关键字文本框都提供关键字提示下拉列表框，这样用户就可以根据以往用户所输入的关键字更加准确地定制自己的关键字，从而在很大程度上提高了要查询内容被高效、准确查询出来的几率。同样，在制作明日知道网站的搜索模块也引用了有关键字提示的搜索条，如图 9.18 所示。

那么，类似 Google 的搜索条是如何实现的呢？是不是特别复杂呢？继续向下学习将会寻求到答案。在具体讲解实现过程前，首先讲解制作该搜索条的思路，当用户在关键字文本框中输入内容时，通过文本框的 onkeyup 事件调用来显示关键字下拉列表的方法，该方法通过 Ajax 技术（这里使用 Ajax 的框架 jQuery），应用 GET 方法向服务器发送请求，请求过程中将用户输入的关键字发送给服务器，服

务器经过处理返回所有以该关键字开头的用户曾搜索的关键字及搜索次数和与该关键字匹配的结果个数。下面具体讲解类似 Google 搜索条的实现方法。



图 9.18　类似 Google 搜索引擎的搜索条

（1）建立用于输入关键字的文本框，这里使用 Zend Framework 框架的 formText()视图助手建立。Zend Framework 框架的视图助手，大部分用来生成组件的和有自动转义变量的功能。另外，有些助手用来创建基于路由的 URL 和 HTML 列表以及声明变量。这里所使用的 formText()助手用来生成 HTML 的文本框组件。

formText()视图助手的语法格式如下：

```
formText(string name, string value, array attribs)
```

参数说明如表 9.1 所示。

表 9.1　formText()视图助手的参数说明

参　　数	说　　明
name	必要参数，指定所生成文本框组件的名称
value	可选参数，文本框的内容，默认为 null
attribs	可选参数，以数组形式指定文本框的多个属性，默认为 null

明日知道网站使用 formText()视图助手生成搜索文本框的代码如下：

例程01　代码位置：光盘\TM\09\known\application\modules\default\views\scripts\index\index.phtml

```
echo $this->formText('keywords', '', array('style'=>'width:350px; height:26px; line-height:26px; font-size:15px;
color:#333333; border:0px;', 'onkeyup'=>'showKeywordsList()', 'onkeydown'=>'selectList()', 'autocomplete'=>
'off'))                                                    //通过 formText()视图助手输出查询关键字录入文本框
```

上述代码中，将关键字文本框命名为 keywords，默认值为空串，在属性数组中，通过键名为 style 的数组元素指定文本框样式，通过键名为 onkeyup 的数组元素指定当按键抬起时所触发的事件，通过键名为 onkeydown 的数组元素指定当按键按下时所触发的事件，最后为键名为 autocomplete 的数组元素指定 off 值来去除文本框自身的下拉列表提示。

（2）当用户在文本框中输入关键字时，将调用自定义的 showKeywordsList()方法显示关键字下拉列表，代码如下：

例程02　代码位置：光盘\TM\09\known\application\modules\default\views\scripts\index\index.phtml

```
var index = 0;                                             //关键字列表项索引
var count = 0;                                             //关键字列表总数
function showKeywordsList(){                                   //显示关键字下拉列表
```

```
    var keycode = event.keyCode;                          //用户按键的 ASCII 码
    if(keycode != 40 && keycode != 38){                   //如果不是向上和向下方向键
        index = 0;                                        //关键字索引归 0
        if($("#keywords").val() == "" || $.trim($("#keywords").val()) == ""){
            $("#searchLayer").css("display", "none");     //如果用户输入空字符或空格，则隐藏下拉列表
        }else{
❶            setTimeout("reSearch()", 150);               //否则每隔 150 毫秒调用一次 reSearch()方法
        }
    }
}
```

代码贴士

❶ 这里之所以要间隔 150 毫秒后调用 reSearch()方法，是为了降低连续对服务器请求而带来的负载。

上述代码中，首先使用 JavaScript 的 event 对象的 keyCode 属性获取用户输入字符的 ASCII 码，然后判断该 ASCII 码是否为 40（代表向上方向键）或 38（代表向下方向键），如果不是这两个方向键，并且用户输入的字符不为空串或空格，则间隔 150 毫秒调用 reSearch()方法来显示与用户输入关键字所匹配的关键字搜索记录。其中 reSearch()方法的代码如下：

例程03 代码位置：光盘\TM\09\known\application\modules\default\views\scripts\index\index.phtml

```
function reSearch(){
    if($("#keywords").val() != ""){                   //如果用户输入的关键字不为空
❶        $.get("<?php echo $this->baseUrl('/index/keywords-list/keyword/"+encodeURI($("#keywords"). val())+"')?>",
null, function(data){
                                                      //通过 jQuery 向服务器发送 GET 请求
            if($.trim(data) == ""){                   //如果返回数据为空，则隐藏下拉列表
                $("#searchLayer").css("display", "none");
            }else{
                $("#searchLayer").html(data);               //如果返回数据不为空，则显示下拉列表
                $("#searchLayer").css("display", "block");
❷                count = parseInt($("#totalList").val());   //获取列表中关键字个数
            }
        });
    }
}
```

代码贴士

❶ encodeURI()方法：用来对关键字进行 URL 编码。

❷ parseInt()方法：用来将字符串转换为整型。

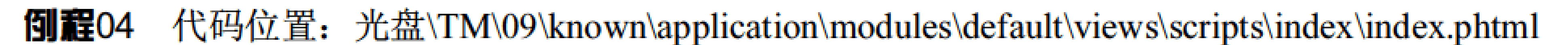
上述代码中，通过 jQuery 的$.get()方法向服务器发送请求，如果返回结果不为空串，则将内容显示在下拉列表中。

说明

通过 Ajax 技术向服务器发送请求，传递的内容应该为 utf-8 编码，如果不是，应该使用函数 iconv()对传递的内容进行转码。

（3）当用户移动键盘的上下方向键时，列表中所选中的项以特殊色显示，并在关键字文本框中显示当前选中项内容。代码如下：





例程04　代码位置：光盘\TM\09\known\application\modules\default\views\scripts\index\index.phtml

```
function selectList(){
    var keycode = event.keyCode;                                           //获取用户输入字符的 ASCII 码
    if(keycode == 40){                                                     //如果用户按向下方向键
        $("#listItem_"+index).css("background-color", "#158CD0");          //更改当前所选项的背景色
        $("#listItem_"+index).css("color", "#FFFFFF");                     //更改当前所选项的前景色
        $("#keywords").val($.trim($("#listItem_li_"+index).html()));       //将当前所选项显示在文本框中
        if(index > 0){
            $("#listItem_"+parseInt(index-1)).css("background-color", "#FFFFFF");
//更改当前项前一项的背景色
                $("#listItem_"+parseInt(index-1)).css("color", "#333333");    //更改当前项前一项的前景色
            }
            if(index < count-1){                                           //如果当前索引小于总关键字数减 1
                index++;                                                   //索引数增 1
            }
        }else if(keycode == 38){                                           //如果用户按上方向键
            if(index > 0){                                                 //如果索引大于 0
                $("#listItem_"+parseInt(index-1)).css("background-color", "#158CD0");
 //更改当前项前一项背景色
                $("#listItem_"+parseInt(index-1)).css("color", "#FFFFFF"); //更改当前项前一项前景色
                $("#keywords").val($.trim($("#listItem_li_"+parseInt(index-1)).html()));
//将当前所选项显示在文本框中
     }
                $("#listItem_"+index).css("background-color", "#FFFFFF");
 //更改当前项背景色
                $("#listItem_"+index).css("color", "#333333");             //更改当前项前景色
                if(index > 1){                                             //如果索引大于 1
                index--;                                                   //索引数减 1
        }
    }
}
```

上述代码首先获取用户输入字符的 ASCII 码，然后判断该值是否为向上方向键或向下方向键所对应的 ASCII 码，如果是则通过 jQuery 的 css()方法改变所选项的前景色和背景色，从而达到关键字列表中内容上下切换的效果。

（4）在上面的讲解中，已经提到客户端通过 GET 方法向服务器发出请求，那么发出请求后做了哪些处理呢？从下述代码中将会找到答案。

例程05　代码位置：光盘\TM\09\known\application\modules\default\controllers\IndexController.php

```
public function keywordsListAction ()
{
❶    header('content-type:text/html; charset=utf-8');                    //设置页面编码为 utf-8
     $keyword = urldecode($this->_request->getParam('keyword'));        //获取用户输入的关键字并进行 URL 解码
❷    $keywordses = $this->_keywordsModel->findByLike($keyword, 10, true);  //调用关键字模型的 findByLink()
方法查询所有以该关键字开头的内容
     $total = 0;                                                         //保存总匹配关键字数
     if ($keywordses != null) {
         $total = count($keywordses);                                    //统计总匹配关键字数
     }
```

```
    $str = '<input type="hidden" name="totalList" id="totalList" value="' . $total . '" />';
    foreach ($keywordses as $k => $key) {                           //通过循环形成下拉列表
        $str .= '<div id="listItem_' . $k . '" style="width:98%; clear:both; background-color:#FFFFFF; cursor:pointer;"
onmouseover="this.style.backgroundColor=\'#158CD0\'; this.style.color=\'#FFFFFF\'" onmouseout= "this.style.
backgroundColor=\'#FFFFFF\'; this.style.color=\'#333333\'" onclick="setText(\'' . $key['keyword'] . '\')">';
        $str .= '<ul>';
        $str .= ' <li id="listItem_li_' . $k . '" style="width:60%; height:18px; line-height:18px; text-align:left; float:left;">' .
$key['keyword'] . '</li>';
        $str .= ' <li style="width:38%; height:18px; line-height:18px; text-align:right; float:right;">' . $key['searchtime'] . '</li>';
        $str .= '</ul>';
        $str .= '</div>';
    }
    echo $str;
    $this->_helper->layout->disableLayout();                        //取消页面布局
    $this->_helper->viewRenderer->setNoRender();                    //取消视图
}
```

代码贴士

❶ header()函数：用于向客户端浏览器发送 HTTP 头信息，这里用来指定页面所使用的编码为 utf-8。

❷ 这里调用关键字模型的 findByLike()方法，获取关键字被搜索最多的 10 条记录。

上述代码为 default 模块中 IndexController 控制器的 keywordsListAction 动作，主要用来获取下拉列表的数据并发送给客户端。

由于上述代码只需要显示下拉列表内容，不需要显示网站头和尾信息，即不需要使用布局，所以使用如下代码取消该动作的页面布局。

```
$this->_helper->layout->disableLayout();
```

注意 使用 header()函数发送 HTTP 头应该在所有页面输出之前。

Zend Framework 中，使用当前控制器对象的_helper 属性的 layout 属性的 disableLayout()方法取消布局。

同样，由于不需要视图，直接通过 echo 语句就可以输出列表中的内容，所以可以将该 Action 所使用的视图通过如下代码取消。

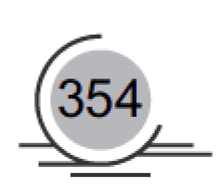
```
$this->_helper->viewRenderer->setNoRender();
```

注意 由于明日知道网站使用 Zend Framework 框架开发，学习本章内容需要有一定 MVC 开发思想和 Zend Framework 框架基础。

9.6.2 使用空格分隔多关键字

用户在使用百度、Google 等互联网搜索工具时，如果对多个关键字同时进行搜索，可以将多个关键字用空格分隔，从而提高查找效率。

明日知道网站中为了方便用户查找，采用分词技术可以同时查找多个关键字，例如用户输入“编



程”和“软件”两个关键字，并且关键字用空格分隔，提交搜索后将查找出所有包含这两个关键字的内容，如图 9.19 所示。

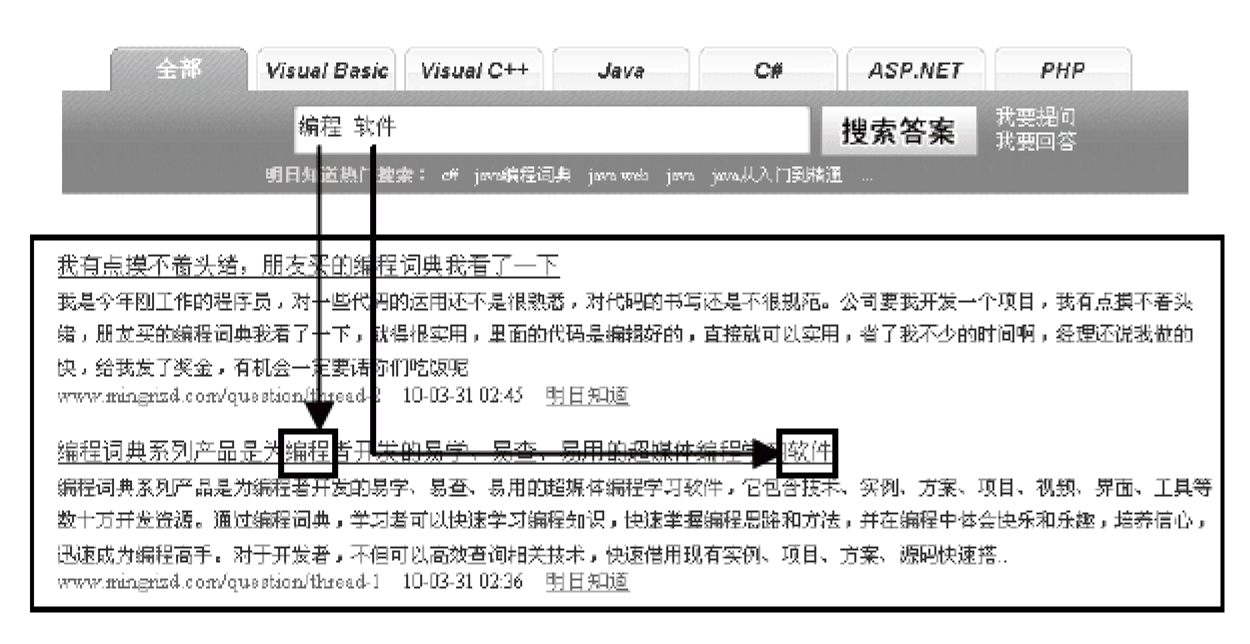


图 9.19　多关键字同时查询

下面就来一起探讨这种实用的分词技术的实现方法。

（1）当用户输入完关键字提交查询内容后，首先在 IndexController 控制器的 searchAction 动作方法中使用 getParam()方法获取用户提交关键字的值，代码如下：

```
$keywordsStr = urldecode($this->_request->getParam('keywords'));
```

Zend Framework 框架中接收页面间传递参数的方法有 3 种，分别为当前控制器对象_request 属性的 getQuery()方法、getPost()方法和 getParam()方法。这 3 个方法的区别是：getQuery()方法用来获取 GET 方法提交的数据，getPost()方法用来获取 POST 方法提交的数据，而通过 getParam()方法既可以获取 GET 方法提交的数据，还可以获取 POST 方法提交的数据。

（2）在问题表（tb_title）模型 Model_DbTable_Title 中建立 findByLike()方法，用来对用户输入的关键字进行模糊查找。代码如下：

例程06　代码位置：光盘\TM\09\known\application\models\DbTable\Title.php

```
public function findByLike ($keywordsStr, $bbstype_id = 0, $orderFlag = null, $page = 1, $pageSize = 10,
$pageRange = 5)
{
❶   $arrayKeyWords = explode(' ', $keywordsStr);                //将多个关键字用空格分隔保存到数组中
    if ($orderFlag == null || $orderFlag == 0) {                //指定排序方式
        $order = 'addtime desc';                                //时间降序
    } else {
        $order = 'browse desc';                                 //时间升序
    }
    $select = $this->getAdapter()->select();                    //生成 Zend_Select 对象
    $select->from($this->_name);                                //为 Zend_Select 对象指定表名
    foreach ($arrayKeyWords as $keywords) {                     //遍历所有被空格分隔的关键字
        if (trim($keywords) != '') {                            //如果关键字不为空串
❷$select->orWhere($this->getAdapter()->quoteInto('title like ?', '%' . $keywords . '%'));
//对主题进行匹配查询
            if ($bbstype_id != 0) {
                $select->where('bbstype_id = ' . $bbstype_id);  //指定查询类别
            }
            $select->orWhere($this->getAdapter()->quoteInto('unhtmlcontent like ?', '%' . $keywords . '%'));
                                                                //对内容进行匹配查询
```

```
                if ($bbstype_id != 0) {
                    $select->where('bbstype_id = ' . $bbstype_id);     //指定查询类别
                }
            }
        }
        $select->order($order);                                         //指定查找结果排序方式
        $adapter = new Zend_Paginator_Adapter_DbSelect($select);    //指定分页适配器
        $paginator = new Zend_Paginator($adapter);                  //生成 Zend_Paginator 分页对象
$paginator->setItemCountPerPage($pageSize)->setCurrentPageNumber($page)->setPageRange($pageRange);
                                                                        //指定分页参数
        return $paginator;                                              //将 Zend_Paginator 对象作为结果返回
}
```

代码贴士

❶ explode()函数：用来将字符串用指定子串分隔，并保存到数组中。

❷ $this->getAdapter()->quoteInto()方法：用来防止非法字符注入。

（3）在 searchAction 动作方法中，通过 tb_title 表模型实例的对象调用 findByLike()方法来获取 Zend_Paginator 对象对查询结果进行分页显示。代码如下：

```
$paginator = $this->_titleModel->findByLike($keywordsStr, $typeid, null, $page, $pageSize);
```

（4）在 searchAction 动作的视图中，通过 foreach 循环语句输出查询结果，代码如下：

```
<?php foreach ($this->paginator as $search):?>
<!---省略 HTML 代码，用来显示查询结果-->
<?php endforeach;?>
```

（5）为了将查询结果进行分页显示，还需要建立分页导航。Zend_Paginator 的分页控制器对象的视图提供了多个便于开发人员建立分页的属性，如表 9.2 所示。

表 9.2　Zend_Paginator 分页控制器对象的视图常用属性

属　　性	说　　明
first	首页页码号
firstItemNumber	当前页第一条记录绝对编号
firstPageInRange	分页导航中首页页码号
current	当前页的页号
currentItemCount	当前页记录总数
itemCountPerPage	每页显示的记录数
last	尾页页码号
lastItemNumber	最后一条记录号
lastPageInRange	分页导航中最后一页页码号
next	下一页页码号
pageCount	总页数
pagesInRange	根据所使用导航条样式，返回页码数组
previous	上一页页码号
totalItemCount	记录总数

了解分页控制器视图的主要属性后，下面开始讲解建立分页导航的步骤。首先需要在 search.phtml 视图文件中所要显示视图导航的位置通过视图对象的 paginationControl()方法指定分页导航的属性，代码如下：

```
<?php echo $this->paginationControl($this->paginator, 'Sliding' ,'searchPaginationControl.phtml', array('keywords' =>
urlencode($this->keywordsStr)))?>
```

视图对象的 paginationControl()方法的语法格式如下：

```
paginationControl(Zend_Paginator paginator, string style, string paginationController, array params)
```

参数说明如表 9.3 所示。

表 9.3 视图对象的 paginationControl()方法的参数说明

属　性	说　明
paginator	必要参数，指定 Zend_Paginator 对象
style	可选参数，指定导航条样式，具体样式说明请参见表 9.4
paginationController	可选参数，指定分页控制器视图文件名
params	可选参数，以数组形式指定要传递给分页控制器视图的变量名称和值

表 9.4 导航条样式说明

参　数	说　明
All	显示所有页码链接
Elastic	类似 Google 所使用的导航条
Sliding	类似 Yahoo 所使用的导航条

最后还需要建立导航视图，明日知道网站中导航视图代码如下：

例程07 代码位置：光盘\TM\09\known\application\modules\default\views\scripts\searchPaginationControl.phtml

```
<ul>
    <li class="li1">每页 <?php echo $this->itemCountPerPage?> 条/共查找到 <?php echo
$this->totalItemCount?> 条</li>
    <li class="li_cell"></li>
    <?php if($this->totalItemCount>0):?>
        <?php foreach ($this->pagesInRange as $page):?>
    <li onclick="javascript:window.location.href='<?php echo $this->baseUrl('/index/search/keywords/'. $this->
keywords.'/page/'.$page)?>';" class="<?php if($this->current==$page):?>li2_current<?php else:?>li2<?php
endif;?>"><a href="<?php echo $this->baseUrl('/index/search/keywords/'.$this->keywords.'/page/'.$page)?>"
class="<?php if($this->current==$page):?>a2<?php else:?>a1<?php endif;?>"><?php echo $page?></a></li>
    <li class="li_cell"></li>
        <?php endforeach;?>
        <?php if($this->last > 5):?>
    <li onclick="javascript:window.location.href='<?php echo $this->baseUrl('/index/search/keywords/'. $this->
keywords.'/page/'.$this->last)?>';" class="li3"><a href="<?php echo $this->baseUrl('/index/search/keywords/'.
$this-> keywords.'/ page/'.$this->last)?>"class="a1">...<?php echo $this->last?></a></li>
        <?php endif;?>
    <?php endif;?>
</ul>
```

用户使用明日知道网站查询内容，将在查询结果页面显示如图 9.20 所示的分页导航条。

每页 20 条/共查找到 110 条 1 2 3 4 5

图 9.20　多关键字同时查询

9.6.3　关键字描红

使用互联网搜索引擎时，当用户在搜索条中输入完关键字并提交查询信息后，会在查询结果显示页面查看到所有输入的查询关键字都以红色显示，这实质就是关键字的描红输出。通过关键字的描红技术，可以使查询结果更醒目，便于用户进一步筛选。明日知道网站的查询结果描红输出的效果如图 9.21 所示。

我有点摸不着头绪，朋友买的编程词典我看了一下
我是今年刚工作的程序员，对一些代码的运用还不是很熟悉，对代码的书写还是不很规范，公司要我开发一个项目，我有点摸不着头绪，朋友买的编程词典我看了一下，就得很实用，里面的代码是编辑好的，直接就可以实用，省了我不少的时间啊，经理还说我做的快，给我发了奖金，有机会一定要请你们吃饭呢
www.mingrizd.com/question/thread-2　10-03-31 02:45　明日知道

编程词典系列产品是为编程者开发的易学、易查、易用的超媒体编程学习软件
编程词典系列产品是为编程者开发的易学、易查、易用的超媒体编程学习软件，它包含技术、实例、方案、项目、视频、界面、工具等数十万开发资源。通过编程词典，学习者可以快速学习编程知识，快速掌握编程思路和方法，并在编程中体会快乐和乐趣，培养信心，迅速成为编程高手。对于开发者，不但可以高效查询相关技术，快速借用现有实例、项目、方案、源码快速搭..
www.mingrizd.com/question/thread-1　10-03-31 02:36　明日知道

图 9.21　查询结果的关键字描红输出

下面将介绍 PHP 中搜索模块关键字描红输出的制作方法。在详细介绍描红的制作步骤前，首先介绍一下描红的实现思路：当用户提交查询关键字到搜索页面后，通过当前控制器对象的_helper 属性的 getParam()方法获取用户提交的关键字，然后使用 PHP 提供的 explode()函数将提交关键字以空格作为分界符保存到数组中，并将该数组保存到视图变量中，最后定义一个视图助手，用于将该数组中所有关键字描红输出。具体实现过程如下：

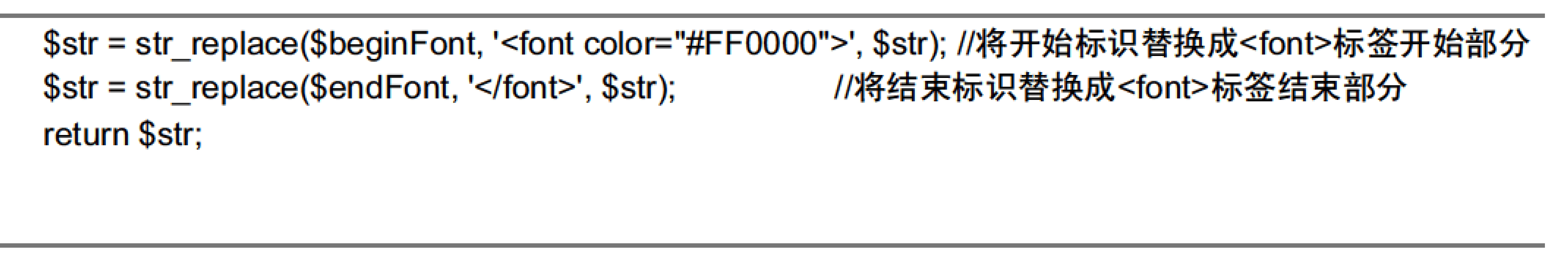
（1）接收用户提交的关键字，并将关键字保存到数组中，然后将该数组赋给视图变量，以便在视图层能够使用该数组，代码如下：

```
$keywordsStrArray = explode(' ', $keywordsStr);                //将关键字字符串用空格分割保存到数组中
$this->view->assign('keywordsStrArray', $keywordsStrArray);    //将数组赋给视图变量 keywordsStrArray
```

（2）在 views 文件夹的 helpers 子文件夹下建立 SetRed.php 文件，在该文件中建立用于将关键字描红输出的自定义助手 Zend_View_Helper_SetRed，代码如下：

例程08　代码位置：光盘\TM\09\known\application\modules\default\views\helpers\SetRed.php

```
class Zend_View_Helper_SetRed
{
    public function setRed ($arrayKeywords, $str)
    {
        $beginFont = '︶';                                    //关键字开始标识
        $endFont = '︵';                                      //关键字结束标识
        foreach ($arrayKeywords as $keyword) {                //遍历所有关键字
            $array = array();
❶           preg_match_all('/' . $keyword . '/i', $str, $array);   //查询与关键字匹配的内容，并保存到数组中
            $arrayKey = array_unique($array[0]);               //去掉重复数组元素
            foreach ($arrayKey as $key) {                      //遍历所有匹配关键字
                $str = str_replace($key, $beginFont . $key . $endFont, $str); //将查询结果中匹配关键字的前后
分别加上关键字开始标识和结束标识
            }
        }
```



```
        $str = str_replace($beginFont, '<font color="#FF0000">', $str); //将开始标识替换成<font>标签开始部分
        $str = str_replace($endFont, '</font>', $str);            //将结束标识替换成<font>标签结束部分
        return $str;
    }
}
```

代码贴士

❶ preg_match_all()函数的语法格式为 preg_match_all(string pattern, string subject, array matches, [.int flags])，该函数用于在 subject 中搜索与 pattern 给出的正则表达式匹配的内容，并将结果以 flags 指定的顺序放到 matches 中。

上述所定义的视图助手包含关键字数组和要描红的字符串两个参数。该视图助手首先通过 foreach()语句遍历关键字数组，然后使用 preg_match_all()函数提取要描红字符串中所有与关键字相匹配的字符串，并将匹配的字符串保存在一个数组中，去掉重复元素后，将该数组所有元素的开始和结束加上一个标识，最后将所有开始标识替换为，结束标识替换成，最终达到在所要描红字符串的所有与关键字匹配子串的前后加上<font>标签，这样可以达到在输出该字符串时，字符串中的匹配关键字描红输出的效果。

说明

Zend Framework 框架中，自定义视图助手一般存放在 views 目录下的 helpers 子目录中，视图助手文件应以大写开头，视图文件中的类名要以 Zend_View_Helper_作为前缀，然后紧接与视图文件名相同的字符作为完整类名，视图文件中包含一个与视图文件名相同的方法，该方法首字母要小写。

（3）在 search.phtml 文件中通过上面介绍的视图助手将查询结果中的关键字进行描红输出，代码如下：

```
echo $this->setRed($this->keywordsStrArray, htmlspecialchars($this->substr($search['unhtmlcontent'], 500)))
```

9.6.4　制作在线编辑器

以往用 HTML 提供的文本域对文本进行内容编辑，一般是通过 JavaScript 在要编辑文本的前后增加一些标记，然后在输出这些内容时，将这些标记替换成相应的 HTML 标记（如<font>、<style>等），这样在显示内容时，就可以出现相应的特效。但是，这种编辑器并不能达到所见即所得的效果，目前已经有很多组织为开发人员提供了在线文本编辑工具，而且功能也非常强大，其中 FckEditor 就是较著名的在线编辑器之一，开发人员只需简单的配置，就可以在项目中应用该编辑工具。不过这些编辑器由于功能较强大，可能存在访问页面时加载速度慢、针对性不强等缺陷。

明日知道网站在发表问题和回复问题时所使用的在线编辑器是通过 JavaScript 编写的，具备文字排版、更改文字样式、插入文字链接和插入图片等功能，完全适合论坛、博客和后台管理中的内容编辑工作。图 9.22 为明日知道网站所使用的在线编辑器。

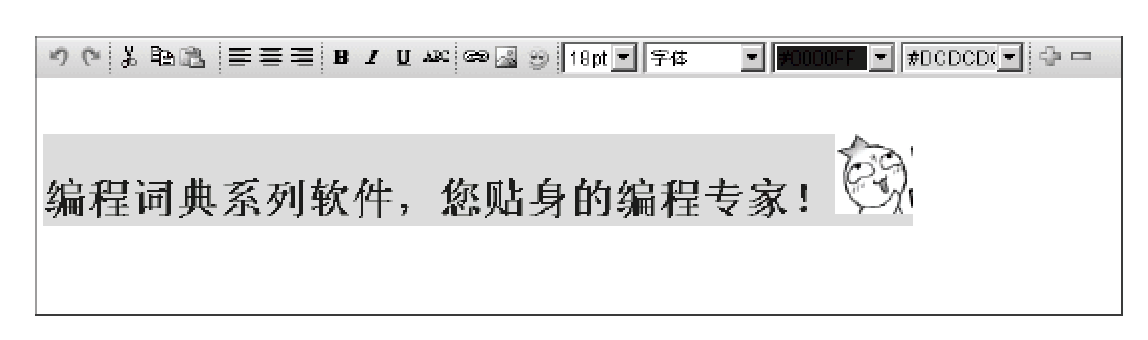


图 9.22　在线编辑器

在讲解在线编辑器具体实现步骤前，首先讲解制作一个在线编辑器的基本思路。明日知道网站所使用编辑器的编辑区域实质为一个<iframe>内嵌标签，默认情况下<iframe>标签是不可编辑的，需要设置<iframe>的 document 属性的 designMode 属性的值为 on，来开启该标签的可编辑模式，然后使用<iframe>的 document 属性的 executeCommand()方法执行相应的命令标识即可，其中常用的命令标识如表 9.5 所示。

表 9.5　常用命令标识符

参　　数	说　　明
BackColor	设置或获取当前选中区的背景颜色
Bold	切换当前选中区的粗体显示与否
CreateLink	在当前选中区上插入超链接，或显示一个对话框允许用户指定要为当前选中区插入的超链接的 URL
Cut	将当前选中区复制到剪贴板并删除
Paste	用剪贴板内容覆盖当前选中区
Copy	将当前选中区复制到剪贴板
Delete	删除当前选中区
FontName	设置或获取当前选中区的字体
FontSize	设置或获取当前选中区的字体大小
ForeColor	设置或获取当前选中区的前景（文本）颜色
Italic	切换当前选中区斜体显示与否
JustifyCenter	将当前选中区所在格式化块置中
JustifyLeft	将当前选中区所在格式化块左对齐
JustifyRight	将当前选中区所在格式化块右对齐
InsertImage	用图像覆盖当前选中区
Undo	取消
Redo	重做
underline	切换当前选中区的下划线显示与否
strikethrough	删除线

了解在线编辑器的基本思路后，下面将具体讲解在线编辑器的制作步骤。

（1）建立在线编辑器 UI。在线编辑器是通过 JavaScript 实现的，所以将 HTML 代码作为字符串，然后使用 JavaScript 的 document 对象的 write()方法输出到浏览器即可。由于代码所占篇幅较长，这里只给出关键代码，具体如下：

例程09　代码位置：光盘\TM\09\known\public_html\js\LzhEditor\LzhEditor.js

```
LzhEditor.prototype.Create = function() {
    this.editorStr += "<div id=\"faceLayer\" style=\"position: absolute; width:445px; z-index: 1; border:1px solid
#77B7DD; background-color:#F2F9F9; clear:both;    display:none;\" onmouseleave=\"this.style.display= 'none'\">";
    ...//省略代码请详见本书附带光盘，该段代码主要用于构建编辑器的控制面板
    this.editorStr += "<iframe id=\"editor\" name=\"editor\" width=\"100%\" height=\""
                + this.height + "\" scrolling=\"auto\" frameborder=\"0\"></iframe>";   //通过 iframe 建立编辑区域
    this.editorStr += "<input type=\"hidden\" id=\"" + this.fieldName
                + "\" name=\"" + this.fieldName + "\" value=\"\" />";   //通过隐藏域保存用户在编辑器输入的内容
    this.editorStr += "</div>";
    document.write(this.editorStr);
```



```
    window.frames["editor"].document.open();                      //打开<iframe>
    window.frames["editor"].document.write("<BODY style=\"PADDING-RIGHT: 5px; PADDING-LEFT: 5px;
FONT-SIZE: 12px; PADDING-BOTTOM: 5px; MARGIN: 0px; PADDING-TOP: 5px\">"+ this.value + "</BODY>");
//将<body>标签写入 iframe
    window.frames["editor"].document.close();                     //关闭<iframe>
❶   window.frames["editor"].document.designMode = "on";           //开启<iframe>的编辑模式
    window.frames["editor"].focus();                              //使<iframe>获得焦点
    window.frames["editor"].document.onkeydown = function() {     //为<iframe>添加 onkeydown 事件
        if (window.frames["editor"].event.keyCode == 13) {        //如果是回车
            window.frames['editor'].document.selection.createRange().pasteHTML('<br><!---->');
  //<br>后必须有内容才能换行，所以<br>后又加了 HTML 的注释，这样既可以实现换行，还不会显示多余内容
            window.frames['editor'].event.returnValue = false;
        }
    };
};
```

代码贴士

❶ 通过设置<iframe>标签 document 属性的 designMode 属性的值为 on，来开启<iframe>标签的编辑模式。

上述代码中，使用<iframe>标签作为在线编辑器的编辑区域，默认情况下<iframe>标签是不可编辑的，所以需要开启<iframe>标签的编辑模式。在开启<iframe>标签的编辑模式前，需要将<body>标签写入到<iframe>内部来设置编辑区域边框间距、文字大小等属性。向<iframe>中写入<body>标签，首先需要调用<iframe>标签的 document 属性的 open()方法打开<iframe>，然后再通过该属性的 write()方法将<body>标签写入到<iframe>内部，最后使用该属性的 close()方法关闭<iframe>标签。

（2）建立完在线编辑器的躯壳后，下一步就需要为编辑器注入灵魂。注入灵魂的过程实质就是让在线编辑器控制面板真正具有排版、设置字体样式等功能，即为控制面板中各个图标按钮的 onClick 事件设置相应的方法。为了提高代码重用率，这里定义一个名为 LzhEditorFormat()的方法，该方法中有两个参数，分别为要执行的命令标识（即表 9.5 所列出的命令标识）及执行该标识所带的参数。该方法的代码如下：

例程10　代码位置：光盘\TM\09\known\public_html\js\LzhEditor\LzhEditor.js

```
function lzhEditorFormat(hc, pa) {
    window.frames["editor"].focus();                                  //编辑器获得焦点
    window.frames["editor"].document.selection.createRange();         //创建编辑区域
    if (pa == "") {
        window.frames["editor"].document.execCommand(hc, false);      //不需要指派参数
    } else {
        window.frames["editor"].document.execCommand(hc, false, pa);  //需要指派参数
    }
}
```

通过上述代码可知，通过<iframe>标签的 document 属性的 execCommand()方法执行命令标识，该方法的语法格式如下：

```
bSuccess = object.execCommand(sCommand [, bUserInterface] [, vValue])
```

参数说明如表 9.6 所示。

表 9.6　execCommand()方法的参数说明

参　数	说　明
sCommand	必要参数，所要执行的命令标识
bUserInterface	可选参数，boolean 型，默认为 false，即不显示用户接口
vValue	可选参数，执行命令标识，所带参数。例如设置字体大小时，字号通过该参数指定

（3）通过上述两步讲解，相信读者已经了解制作一个在线编辑器的基本思路，最后还需要讲解如何在程序中调用该编辑器。

由于该编辑器被封装在一个独立的 js 文件中，所以首先应该通过<script>标签的 src 属性包含该 js 文件，然后在 form 表单要显示编辑器的位置使用如下代码引入编辑器。

```
var lzhEditor = new LzhEditor(name, width, height ,value, baseUrl, uploadFileUrl);    //实例编辑器
lzhEditor.Create();                                                                    //创建编辑器
```

通过上述代码就可以将编辑器显示在表单区域内，但当用户提交表单前，还需要通过如下代码将编辑器中的内容赋给隐藏域，表单提交时不能直接将编辑器中的内容提交给服务器，而是间接地通过隐藏域实现。

```
lzhEditor.Submit();                                              //将编辑器中的内容赋给隐藏域
```

以上就是制作一个在线编辑器的全部过程，最后在总结下其开发思路：首先开启<iframe>标签的编辑模式，然后通过 execCommand()方法执行对应的命令标识即可对<iframe>中的文本进行编辑。

9.7　MVC 框架结构搭建

明日知道网站采用 Zend Framework 框架技术开发，所以首先应该设计并搭建出稳定、扩展性强的框架结构。为了能够掌握 Zend Framework 框架的搭建方法，下面通过一个简单的实例对此进行介绍。

9.7.1　MVC 开发模式介绍

MVC 分别为英文单词 model、view 和 controller 的首字母，中文含义为“模型-视图-控制器”，这种开发模式是现阶段主流的项目开发模式之一，可使整个项目的各个层次独立出来，这样程序开发人员和美工人员可以对项目进行同步开发，而且也为项目的进一步功能扩展和日后在生产环境下的维护工作提供了极大方便，所以 MVC 开发模式也是现阶段各大软件公司极力提倡的开发模式，是程序开发人员的必修内容。图 9.23 为

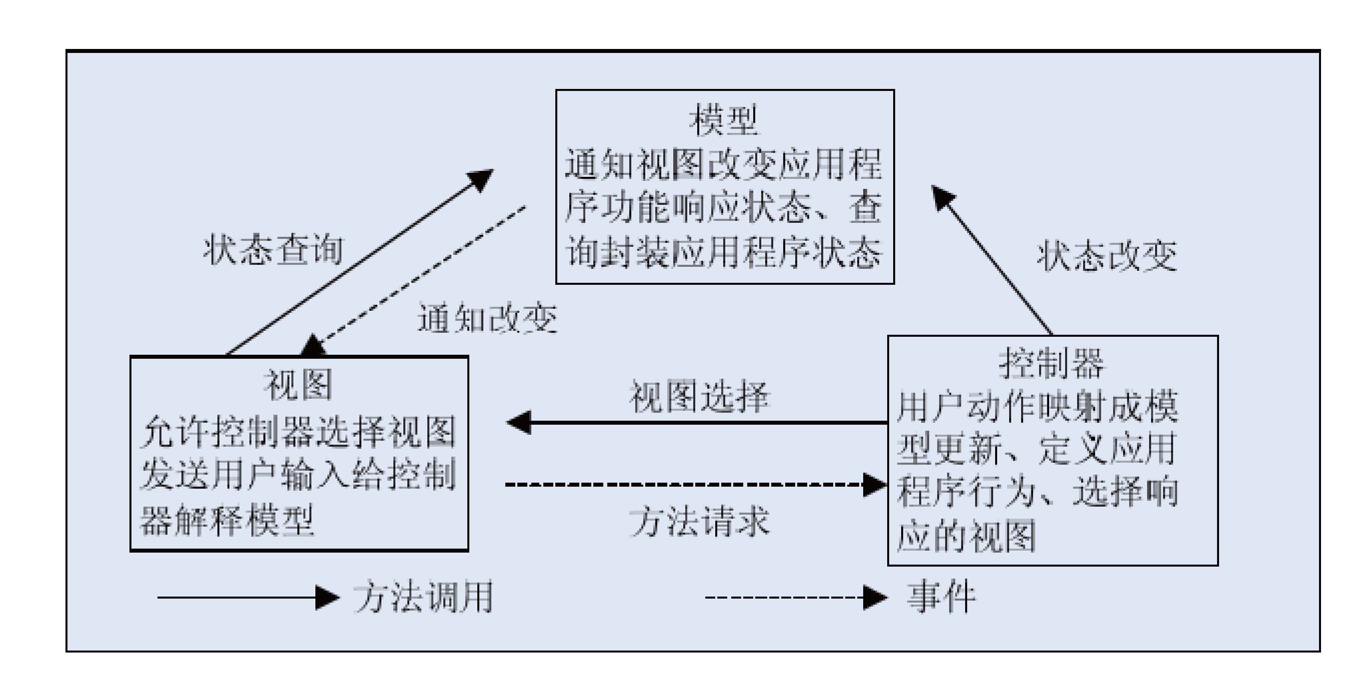
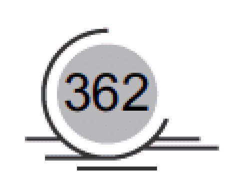


图 9.23　MVC 模式架构图



典型的 MVC 模式架构图。

9.7.2　Zend Framework 框架的 MVC 目录结构

与其他主流 PHP 框架相比，Zend Framewok 是一款相对灵活的 MVC 框架，其搭建方式有多种，其中明日知道网站的 MVC 框架结构如图 9.24 所示。从图中可以看出，该结构是一种多模块结构，即将前台和后台独立成两个模块，从而更有利于多人协作开发。

9.7.3　Zend Framework 框架的 MVC 结构创建过程

了解明日知道网站的框架结构图后，下面具体按步骤讲解使用 Zend Framework 建立多模块 MVC 框架结构的搭建过程。为了便于学习，这里给出一个最基本的 Zend Framework 框架的流程，如图 9.25 所示，之后开发人员就可以在此基础上进行扩充。

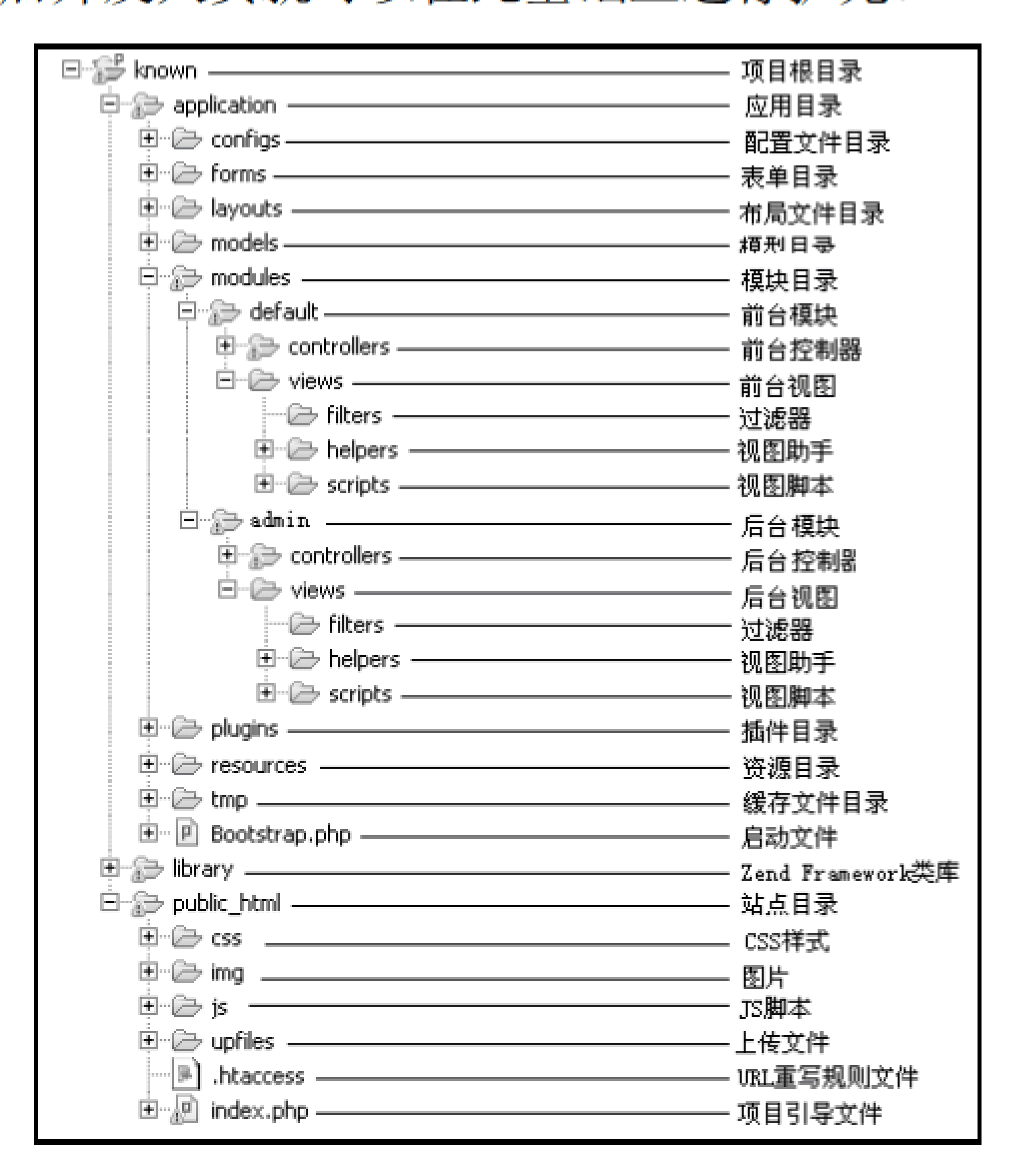


图 9.24　明日知道网站框架结构

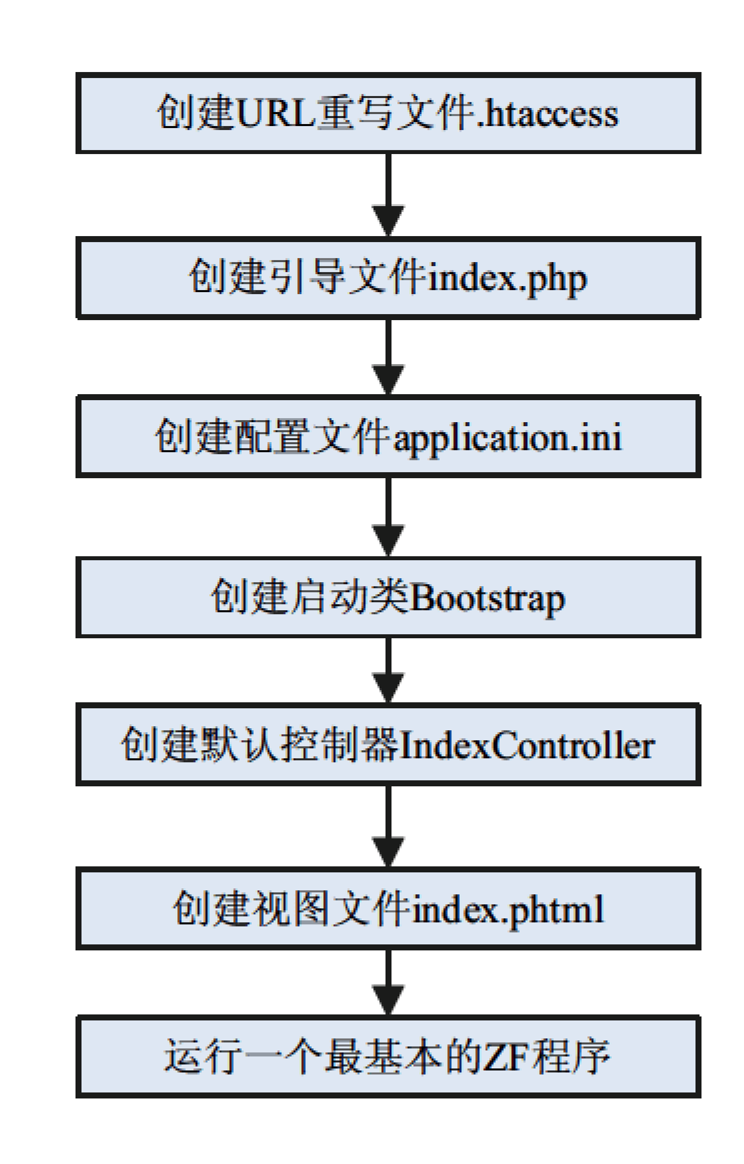


图 9.25　最基本的 Zend Framework 程序开发步骤

（1）在 public_html 目录下建立 URL 重写文件.htaccess，并在该文件中输入 URL 重写规则，如下所示：

```
#开启 URL 重写
RewriteEngine on
#除扩展名为.js、.css、.gif、.jpg、.png、.bmp 的文件外，访问其他文件都转向到 index.php 引导文件
RewriteRule !\.(js|css|gif|jpg|png|bmp)$ index.php
```

说明

由于.htaccess 文件没有文件主名，所以在 Windows 系统下无法直接命名，下面介绍两种用户创建.htaccess 文件的方法。

方法一：通过 Windows 系统的 copy con 命令创建，如图 9.26 所示。创建完成后使用 Ctrl+Z 组合键退出编辑模式。



图 9.26 通过命令提示符创建.htaccess 文件

方法二：通过“记事本”文本编辑工具将文件另存为后缀名为.htaccess 的文件，这里应该注意保存文件时需要选择文件类型为所有文件。

（2）在 public_html 目录下创建 index.php 引导文件，代码如下：

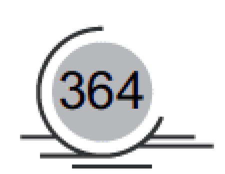
```
defined('APPLICATION_PATH') || define('APPLICATION_PATH', realpath(dirname(__FILE__) . '/../application'));
                                                                    //应用路径
defined('APPLICATION_ENV') || define('APPLICATION_ENV', getenv('APPLICATION_ENV') ? getenv ('APPLICATION_
ENV') : 'project');                                                 //应用环境
$arrayIncludePath = array('.', realpath(dirname(__FILE__) . '/../library'));     //指定工程包含目录
set_include_path(implode(PATH_SEPARATOR, $arrayIncludePath));       //将指定路径包含到工程中
require_once 'Zend/Application.php';                                //包含 Application.php 文件
$application = new Zend_Application(APPLICATION_ENV, APPLICATION_PATH . '/configs/application.ini');
                                                                    //实例化 Zend_Application 类
$application->bootstrap()->run();                                   //调用启动文件并运行项目
```

上述代码中，首先定义两个常量 APPLICATION_PATH 和 APPLICATION_ENV，分别用来保存应用路径和应用环境名，这样在工程的其他模块中就可以直接使用。然后将当前目录地址“.”和 Zend Framework 类库地址作为数组元素保存在名为$arrayIncludePath 的数组中，并使用 implode()函数将上述路径用 PATH_SEPARATOR 常量连接，最后使用 set_include_path()方法将路径导入到工程中，从而在工程中就可以直接找到这些路径下的文件。

Zend Framework 类库被导入到工程后，就可以使用 require_once 等文件包含语句包含 Zend 目录下的 Application.php 文件，之后实例化 Zend_Application 类，并通过引用该类的 bootstrap()方法调用启动类，最终再通过调用 run()方法运行工程。

（3）在第（2）步实例 Zend_Application 类时，为类的构造函数传入一个名为 application.ini 的文件，该文件即为工程的配置文件，本步将介绍该文件的创建过程，其中一个基本的配置文件如下：

```
 [project]
#设置错误级别
phpSettings.display_startup_errors = 1
phpSettings.display_errors = 1
#设置时区
phpSetting.date.timezone = 'Asia/Shanghai'
#配置启动类
bootstrap.path = APPLICATION_PATH "/Bootstrap.php"
bootstrap.class = "Bootstrap"
#配置工程模块
resources.frontController.moduleDirectory = APPLICATION_PATH "/modules"
```



配置文件存放位置和名称并不固定，只要与实例化 Zend_Application 类传递的参数相一致即可。

（4）在 application.ini 配置文件中指定启动类的位置后，工程就可以找到启动文件 Bootstrap.php，其关键代码如下：

```
clclass Bootstrap extends Zend_Application_Bootstrap_Bootstrap
{
//基本的启动类可以不进行任何操作
}
```

从上述代码可知，启动 Bootstrap 继承自类 Zend_Application_Bootstrap_Bootstrap，该类已经实现最基本的启动配置工作，所以在启动类中不需包含任何代码就可以运行最基本的 Zend Framework 应用。

（5）完成以上步骤后，就可以创建默认控制器 IndexController，同样一个最基本的控制器应该包含一个首页动作 indexAction，代码如下：

```
class IndexController extends Zend_Controller_Action
{
    public function indexAction ()                              //默认动作
    {
        $this->view->testStr="Hello ZF!";                       //为视图变量赋值
    }
}
```

说明　使用 Zend Framework 时，控制器名应该使用 Controller 结束，动作名应以 Action 结束。

（6）完成默认控制器及首页动作的创建后，就可以创建视图文件 index.phtml，为了查看效果，在视图文件中仅输出首页动作中指定的视图变量的值，代码如下：

```
echo $this->testStr;
```

（7）假设上述工程所在目录为 Apache 默认主目录的 test 子目录下，就可以通过如下 URL 进行访问：

☑ http://localhost/test/public_html

☑ http://localhost/test/public_html/index

☑ http://localhost/test/public_html/index/index

正常情况下，Zend Framework 的访问 URL 应该为“域名+模块名+控制器名+动作名”，那么为什么上述 URL 省略了部分内容也可以访问呢？这是因为在 Zend Framework 中，默认模块名、默认控制器名和首页动作名可以省略。

运行上述工程，可以在浏览器中输出“Hello ZF!”。

9.8　首 页 设 计

网站首页的内容应该突出网站主题及特色，在网站的生命周期中起着非常重要的作用，本节将以明日知道网站为例讲解如何设计合理的、适合网站特点的首页。

9.8.1 首页概述

设计网站首页要贴切网站主题、突出网站特色，而非复杂就可以。考虑到明日知道网站主要用于为编程相关工作人员提供技术问题搜索和技术交流，借鉴互联网一些知名的搜索工具，设计出如图 9.27 所示的首页界面。明日知道网站的首页主要由网站 Logo、语言导航条、搜索区、友情链接和版权声明等部分组成，页面简洁大方、突出核心，便于用户使用。

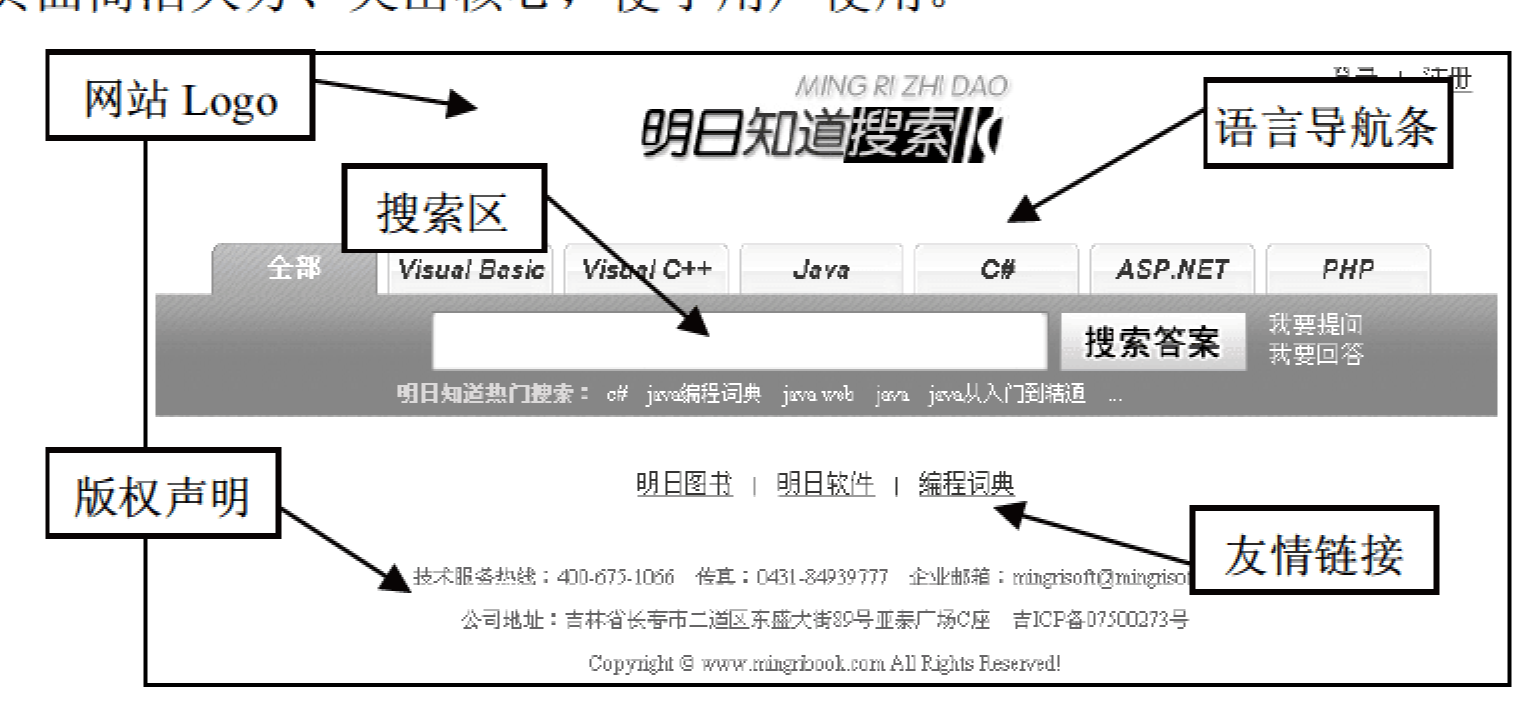


图 9.27　明日知道网站首页

9.8.2 首页技术分析

明日知道网站首页主要涉及两点较为关键的技术，分别为类似 Google 搜索条的制作和语言类别选项卡的制作，其中类似 Google 搜索条的制作过程已经在 9.6 节中进行讲解，这里不再赘述，下面重点对语言类别选项卡的实现过程进行技术分析。

与 Visual Basic、C#等一些可视化编程语言相比，网页视图层的一些组件和特效的制作需要通过编码实现，而非简单的拖曳，这样制作出的组件更具灵活性。制作页面特效一般通过 JavaScript、Ajax 等技术实现。

制作明日知道网站语言类别选项卡的基本思想是：首先从数据库中提取所要显示语言类别的名称，并通过<div>标签显示在同一行，然后为这些<div>标签指定背景图片，最后定义如下方法（这里用 jQuery 实现）控制选项卡之间的切换。

例程11　代码位置：光盘\TM\09\known\application\modules\default\views\scripts\index\index.phtml

```
function changeSearchType(x, y)
{    //该方法中的两个参数 x、y 分别为每个选项卡按钮的 ID 和选项卡对应语言的 ID
     $("#st_0").css("background", "url(<?php echo $this->baseUrl('/img/main_dh_btn2.gif')?>)"); //更改全部选项卡背景图片
     $("#st_0").css("color", "#333333");                //更改全部选项卡前景色
     <?php foreach ($this->types as $type):?>
          $("#st_<?php echo $type['tid']?>").css("background", "url(<?php echo $this->baseUrl('/img/main_dh_btn2.gif')?>)");                //更改语言类别选项卡背景图片
          $("#st_<?php echo $type['tid']?>").css("color", "#333333");   //更改语言类别选项卡前景色
     <?php endforeach;?>
     $("#"+x).css("background", "url(<?php echo $this->baseUrl('/img/main_dh_btn1.gif')?>)");  //更改当前选中的选项卡背景图片
```

```
    $("#"+x).css("color", "#FFFFFF");                   //更改当前选中的选项卡的前景色
❶   $("#typeid").val(y);                                //更改隐藏域 typeid 的值为当前选中选项卡对应的语言 ID
}
```

上述代码中，通过 jQuery 的 css()方法更改<div>标签的背景图片和前景色，该方法可以传入两个参数，第一个参数为要更改的 CSS 样式属性名称，如 color、background-color、font-size 等，第二个参数是为属性所指定的值，其中第二个参数可以省略，如果设置第二个参数，则为 CSS 样式重新赋值，否则将获取该 CSS 样式的值。

9.8.3　首页的实现过程

明日知道网站采用 MVC 开发模式，所以其开发过程也需要按模型层、视图层和控制器层的顺序进行。在对首页进行技术分析时已经介绍到，制作该站首页需要从数据库中提取所有语言类别，所以应该在类别表的模型 Model_DbTable_Bbstype 中建立一个用来查询全部类别的方法，这里命名为 findAll()。代码如下：

例程12　代码位置：光盘\TM\09\known\application\models\DbTable\Bbstype.php

```
public function findAll ($isCache = false)
{
    $innerSelect1 = $this->getAdapter()->select();                //所有主题
    $innerSelect1->from('tb_title', 'count(*)')->join($this->_name, 'tb_title.bbstype_id=' . $this->_name . '.id', null)->
where($this->_name . '.id = tid');
    $innerSelect2 = $this->getAdapter()->select();                //今日主题
    $innerSelect2->from('tb_title', 'count(*)')->join($this->_name, 'tb_title.bbstype_id=' . $this->_name . '.id', null)->
where('date(tb_title.addtime)=date(now())')->where($this->_name . '.id = tid');
    $innerSelect3 = $this->getAdapter()->select();                //最后主题 ID
    $innerSelect3->from('tb_title', 'tb_title.id')->join($this->_name, 'tb_title.bbstype_id=' . $this->_name . '.id', null)->
where($this->_name . '.id = tid')->order('tb_title.addtime desc')->limit(1, 0);
    $innerSelect4 = $this->getAdapter()->select();                //最后主题
    $innerSelect4->from('tb_title', 'tb_title.title')->join($this->_name, 'tb_title.bbstype_id=' . $this->_name . '.id', null)->
where($this->_name . '.id = tid')->order('tb_title.addtime desc')->limit(1, 0);
    $innerSelect5 = $this->getAdapter()->select();                //最后主题时间
    $innerSelect5->from('tb_title', 'tb_title.addtime')->join($this->_name, 'tb_title.bbstype_id=' . $this->_name .
'.id', null)->where($this->_name . '.id = tid')->order('tb_title.addtime desc')->limit(1, 0);
    $innerSelect6 = $this->getAdapter()->select();                //总回复
    $innerSelect6->from('tb_reply', 'count(*)')->join($this->_name, 'tb_reply.bbstype_id=' . $this->_name . '.id', null)->
where($this->_name . '.id = tid');
    $innserSelect7 = $this->getAdapter()->select();               //最后回复用户 ID
    $innserSelect7->from('tb_title', 'tb_title.user_id')->join($this->_name, 'tb_title.bbstype_id=' . $this->_name .
'.id', null)->where($this->_name . '.id=tid')->order('tb_title.addtime desc')->limit(1, 0);
    $innserSelect8 = $this->getAdapter()->select();               //最后回复用户昵称
    $innserSelect8->from('tb_user', 'netname')->where('id=(' . $innserSelect7 . ')');
    $select = $this->getAdapter()->select();                      //外层查询
    $select->from($this->_name, array($this->_name . '.id as tid' , $this->_name . '.typename' , $this->_name .
'.description' , $this->_name . '.addtime' , '(' . $innerSelect1 . ') as totaltitle' , '(' . $innerSelect2 . ') as
totaltodaytitle' , '(' . $innerSelect3 . ') as lasttitleid' , '(' . $innerSelect4 . ') as lasttitletitle' , '(' . $innerSelect5 . ') as
lasttitletime' , '(' . $innerSelect6 . ') as totalreply' , '(' . $innserSelect8 . ') as lasttitlenetname'))->order('' .
$this->_name . '.addtime asc');
```

```
    if ($isCache) {                                           //如果缓存查询结果
        if (! $result = $this->_cache->load(strtoupper($this->_name . '_findAll'))) {
//判断缓存文件是否存在或缓存是否过期
            $result = $this->getAdapter()->fetchAll($select);   //如果缓存文件不存在或缓存过期，则重写查找
$this->_cache->save($result, strtoupper($this->_name . '_findAll'));
//写入并生成新缓存
        }
    } else {
        $result = $this->getAdapter()->fetchAll($select);       //不缓存则直接查找
    }
    return $result;                                           //返回结果
}
```

上述代码中查询全部语言类别通过 Zend_Select 并使用子查询及 Zend_Cache 缓存技术实现。

首先介绍如何使用 Zend_Select 进行查询。使用 Zend_Select 进行查询，需要获得数据库适配器对象，然后通过该对象的 select()方法生成 Zend_Select 对象，代码如下：

```
$select = $this->getAdapter()->select();
```

其中，$this 代表当前表的模型，由于该模型继承自 Zend_Db_Table 类，所以通过$this->getAdapter()方法即可获取数据库适配器。获得 Zend_Select 对象后，就可以通过该对象的 from()方法，指定查询的表名、字段名；通过 order()方法指定记录排序方式。

那么如何通过 Zend_Select 进行子查询呢？其原理是：首先定义一个 Zend_Select 对象作为子查询，然后在外层查询中，将该 Zend_Select 对象作为查询结果的一个字段出现，代码如下：

```
$inSelect = $db->select();                                    //内层查询，这里只给出关键代码
$outSelect = $db->select();                                   //外层查询
$outSelect->from(表名, array('字段 1', '字段 2', '('.$inSelect.') as  字段 3' ));//指定外层查询表名和字段名
```

最后介绍如何对查询结果进行缓存输出。Zend Framework 框架中，使用 Zend_Cache 对数据进行缓存来提高程序执行效率。在制作上述查询全部语言类别方法时，为了降低对数据库频繁查询的次数，降低数据库服务器负载，提交程序执行效率，这里使用 Zend_Cache 对查询结果进行缓存。Zend_Cache 通过其静态方法 factory()获得其实例，代码如下：

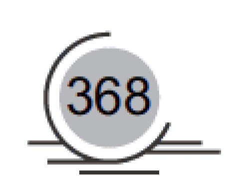
```
$cache = Zend_Cache::factory(string frontendName, string backendName, array frontendOptions, array backendOptions);
```

参数说明如表 9.7 所示。

表 9.7　Zend_Cache 的 factory()方法的参数说明

属　性　值	说　　明
frontendName	必要参数，缓存前端名称
backendName	必要参数，缓存后端名称
frontendOptions	可选参数，缓存前端选项
backendOptions	可选参数，缓存后端选项

为了更加明确地了解 Zend_Cache 的使用方法，下面以本项目为例按步讲解 Zend Framework 中缓存的使用方法。



（1）为了便于以后维护，将缓存过期时间和缓存文件路径在 application.ini 配置文件中指定，代码如下：

```
#缓存时间
cache.leftTime = "3600"
#缓存文件存储目录
cache.cache_dir = APPLICATION_PATH "/tmp/"
```

（2）Zend_Cache 对象应该被项目全局应用，所以 Zend_Cache 对象在启动文件中被初始化，代码如下：

```
$config = $this->getOptions();    //通过启动类的 getOptions()方法获取配置文件配置信息，并保存在多维数组中返回
Zend_Registry::set('config', $config);                                    //注册该多维数组
$frontOptions = array('leftTime' => $config['cache']['leftTime'] , 'automatic_serialization' => true);
                                                                          //缓存前端选项
$backOptions = array('cache_dir' => $config['cache']['cache_dir']);       //缓存后端选项
$cache = Zend_Cache::factory('Core', 'File', $frontOptions, $backOptions); //获取 Zend_Cache 对象
Zend_Registry::set('cache', $cache);                                      //注册 Zend_Cache 对象
```

（3）使用 Zend_Register 的静态方法 set()将缓存对象注册后，就可以在工程的其他模块通过 Zend_Register 的 get()方法获取缓存对象并使用。代码如下：

```
private $_cache;                     //将缓存对象声明为私有成员，这样就可以在模型的其他方法中使用
public function init ()
{
    $this->_cache = Zend_Registry::get('cache');     //获取缓存对象
}
```

（4）在类别表模型的 findAll()方法中，通过缓存对象的 load()方法判断在缓存目录下是否有以该方法参数为缓存 ID 的缓存文件并且不过期，如果存在则从缓存文件中获取要查询的内容，否则从数据库中查询内容并保存在缓存文件中，代码如下：

```
if (! $result = $this->_cache->load(strtoupper($this->_name . '_findAll'))) {//判断缓存文件是否存在或缓存是否过期
    $result = $this->getAdapter()->fetchAll($select);           //如果缓存文件不存在或缓存过期，则重写查找
    $this->_cache->save($result, strtoupper($this->_name . '_findAll'));   //写入并生成新缓存
}
```

9.9　用户模块设计

用户作为网站活动的参与者和维护者，是网站能够生存发展的源动力。明日知道网站由于需要用户提出问题、解决问题，所以一个合理的用户模块显得十分重要。

9.9.1　用户模块概述

明日知道网站的用户模块，主要包括用户注册、登录、用户信息管理等功能。浏览者如果只进行问题查找，则无须注册，如果没有在该站点查找到预期的结果，那么可以将问题发布到明日知道网站

中，其他用户就可以与之探讨、解答，这时就需要用户注册为本站会员。用户访问明日知道网站的流程如图 9.28 所示。

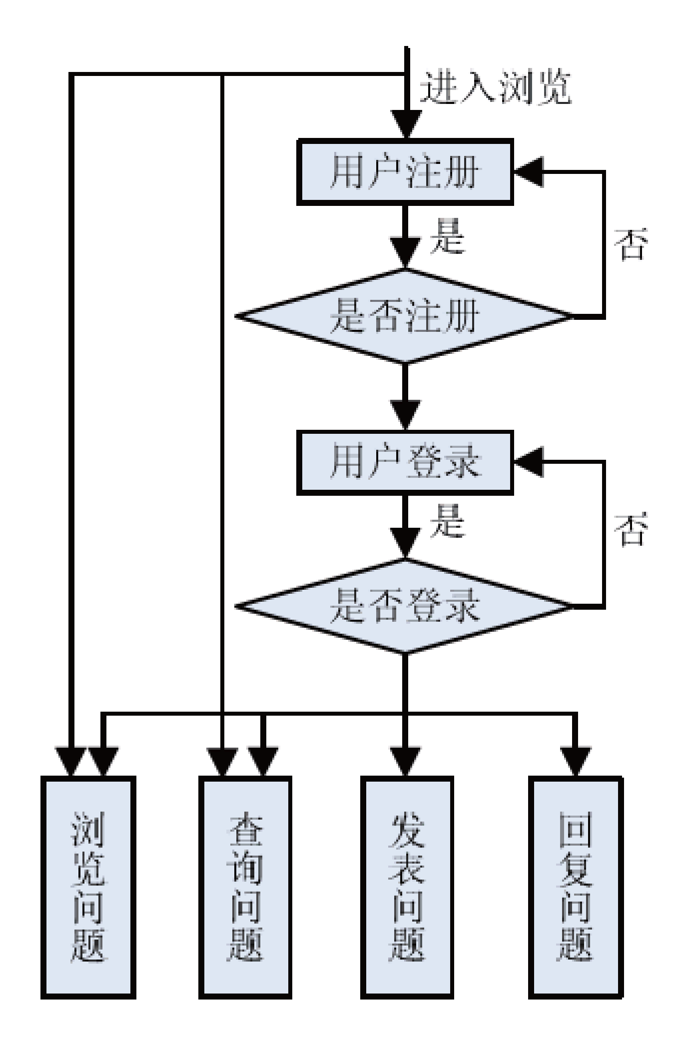


图 9.28　用户访问流程

9.9.2　用户模块技术分析

为了防止非法用户通过恶意程序，采用试探密码的方式登录本站，或连续地向用户表中注入信息，明日知道网站的用户登录和注册表单中都增加了验证码，从而可以有效地预防上述情况发生。

在 Web 项目中，验证码可以有多种形式，如文字验证码、图片验证码、语言验证码等，本站所采用的是 Web 项目中最为常用的验证码形式，即图片验证码，如图 9.29 所示。

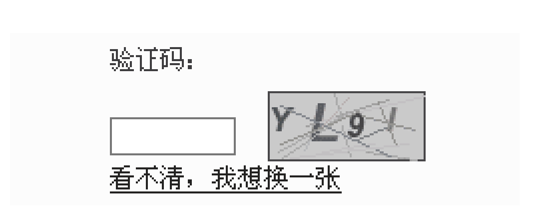


图 9.29　图片验证码

使用 PHP 语言制作验证码图片可以通过 GD2 函数绘制。考虑到验证码的高度、宽度以及图片中文字的字体在不同类型的表单中可能存在差异，在制作明日知道网站验证码时，单独将验证码的实现过程封装成验证码类。那么，验证码类是如何实现的呢？下面对此进行讲解。

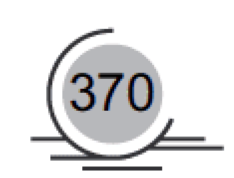
（1）命名该类为 Plugin_Util_ValidateCode，这样命名的原因是如果将该验证码类保存在 application 目录下的 util 子目录中，并将该文件命名为 ValidateCode.php，那么在需要使用该验证码类时，根据 Zend Framework 的设计特点，不需要使用 include 等包含语句，就可以直接使用该验证码类。

定义该验证码类的数据成员，代码如下：

例程13　代码位置：光盘\TM\09\known\public_html\known\application\plugins\Util\ValidateCode.php

```
private $_width;                                    //宽度
private $_height;                                   //高度
private $_codeStr;                                  //验证码
private $_fontType;                                 //设置字体类型，0 表示粗体，1 表示简体
private $_img;                                      //图像句柄
```

（2）定义验证码类的构造方法，实现验证码类的初始化，代码如下：



例程14　代码位置：光盘\TM\09\known\public_html\known\application\plugins\Util\ValidateCode.php

```
public function __construct ($width, $height, $codeStr = '0000', $fontType = 0)
{
    $this->_width = $width;                              //验证码宽度
    $this->_height = $height;                            //验证码高度
    $this->_codeStr = substr($codeStr, 0, 4);            //验证码的值
    $this->_fontType = $fontType;                        //字体
}
```

（3）定义类的私有方法_getColor()，用于生成颜色句柄，代码如下：

例程15　代码位置：光盘\TM\09\known\public_html\known\application\plugins\Util\ValidateCode.php

```
private function _getColor ($x, $y)
{
    $r = mt_rand($x, $y);                                //红
    $g = mt_rand($x, $y);                                //绿
    $b = mt_rand($x, $y);                                //蓝
    return imagecolorallocate($this->_img, $r, $g, $b);  //返回颜色句柄
}
```

（4）定义私有方法_init()，用于创建图片验证码对象，代码如下：

例程16　代码位置：光盘\TM\09\known\public_html\known\application\plugins\Util\ValidateCode.php

```
private function _init ()
{
    $this->_img = imagecreate($this->_width, $this->_height);     //创建图像
}
```

（5）在创建验证码图片对象的基础上创建验证码体，包括设置验证码的填充色、边框、验证码上的文字内容及字体等，其实现代码如下：

例程17　代码位置：光盘\TM\09\known\public_html\known\application\plugins\Util\ValidateCode.php

```
private function _build ()
{
    imagefill($this->_img, 0, 0, $this->_getColor(150, 250));     //为图像填充背景色
imagerectangle($this->_img, 0, 0, $this->_width - 1, $this->_height - 1, $this->_getColor(50, 150));
                                                                  //创建一个矩形作为验证码的边框
    if ($this->_fontType == 0) {                                  //设置验证码文字的字体
        $fontFileName = 'ARIALBI.TTF';
    } else {
        $fontFileName = 'ARIALN.TTF';
    }
    for ($i = 0; $i < strlen($this->_codeStr); $i ++) {          //绘制文字
❶       imagettftext($this->_img, mt_rand(12, 24), 0, ($this->_width) / 4 * $i, mt_rand(20, $this->_height - 5),
$this->_getColor(10, 180), APPLICATION_PATH . '/resources/' . $fontFileName, substr($this->_codeStr, $i, 1));
    }
    for ($i = 0; $i < 15; $i ++) {                               //绘制 15 条干扰线
❷       imageline($this->_img, mt_rand(0, $this->_width), mt_rand(0, $this->_height), mt_rand(0, $this->_width),
mt_rand(0, $this->_height), $this->_getColor(110, 210));
```

```
    }
}
```

代码贴士

❶ imagettftext()方法：用于按指定的字体输出字符串。

❷ imageline()方法：用于绘制直线。

（6）创建 show()方法用于显示图片，代码如下：

例程18　代码位置：光盘\TM\09\known\public_html\known\application\plugins\Util\ValidateCode.php

```
public function show ()
{
    header('content-type:image/png');                    //设置输出图片格式
    $this->_init();                                      //图片初始化
    $this->_build();                                     //建立图片体
    imagepng($this->_img);                               //输出图片
}
```

9.9.3　用户注册的实现过程

通过用户注册，可将用户的基本信息记录下来，便于用户参与站内活动及用户之间的交流。明日知道网站的注册表单如图 9.30 所示，该注册表单主要包括用户的昵称、登录密码、E-mail 地址、联系电话和所在地等信息。

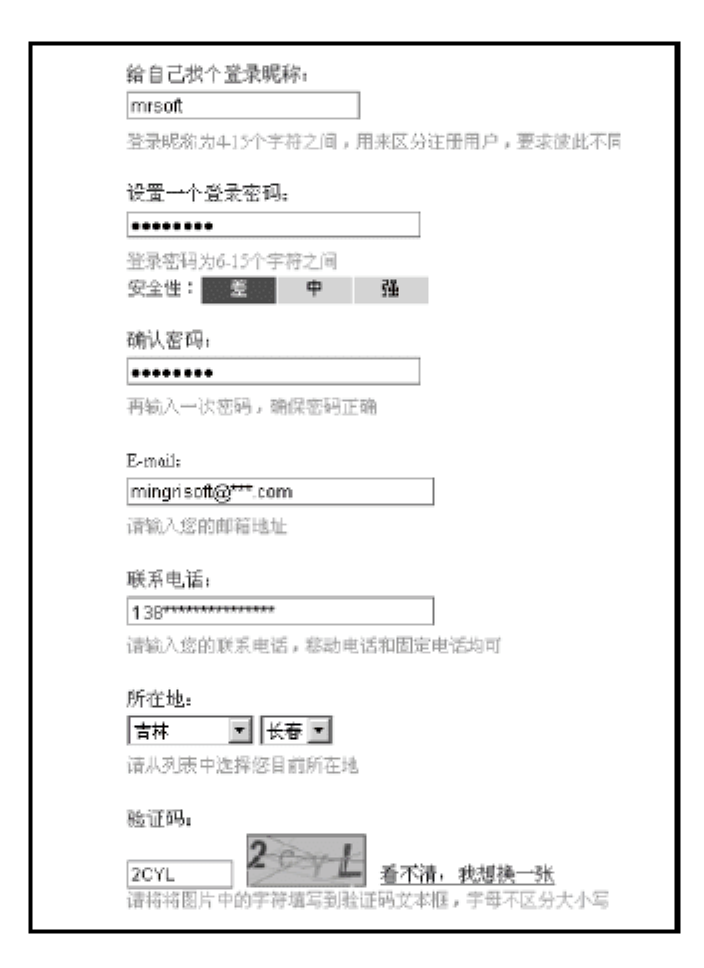


图 9.30　用户注册表单

了解该注册表单的结构后，下面讲解该表单的实现过程。

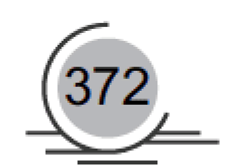
（1）建立用户注册表单。明日知道网站的注册表单是通过 Zend Framework 的表单视图助手实现的，并通过 jQuery 技术对表单数据进行校验。注册表单 UI 的实现代码请参见本书附带光盘。

（2）保存用户注册信息，并向注册用户的邮箱中发送注册成功提示。当用户填写完注册信息，并提交注册表单后，用户注册信息将被提交到用户控制器的注册动作中进行处理，代码如下：

例程19　代码位置：光盘\TM\09known\application\modules\default\controllers\UserController.php

```
if ($this->_request->isPost()) {                          //如果用户通过 POST 方法提交表单
    $request = $this->_request;                           //获取当前控制器对象的 POST 方法
    $arrayProvice = Plugin_Util_ProvinceAndCityFactory::getProvince();   //获取用户所在省份
    $p = $arrayProvice[$request->getParam('p')];
    $arrayCity=Plugin_Util_ProvinceAndCityFactory::getCityByProvinceArrayIndex($request->getParam('p'));
    $c = $arrayCity[$request->getParam('c')];             //获取用户所在市

    $config = Zend_Registry::get('config');               //获取 Zend_Config 对象
    $arraySmtpConfig = array('auth' => 'login' , 'username' => $config['mail']['username'] , 'password' =>
$config['mail']['password']);   //邮件配置信息
    $transport = new Zend_Mail_Transport_Smtp($config['mail']['host'], $arraySmtpConfig);//构建邮件传输对象
❶  $mail = new Zend_Mail('utf-8');                        //构建 Zend_Mail 对象，并设置字符编码
```



```
    $mail->setSubject($config['mail']['subject']);            //邮件主题
    $mail->setBodyHtml(file_get_contents($config['mail']['bodyPath']));          //HTML 邮件内容
    $mail->setFrom($config['mail']['from'], $config['mail']['name']);  //发件人
    $mail->addTo(trim($request->getParam('email')));       //收件人
    try {
        $mail->send($transport);                              //发送邮件
    } catch (Zend_Exception $e) {
        $e->getMessage();
    }
                                                              //保存用户信息到数据库
    $arrayUserInfo = array('netname' => trim($request->getParam('netname')) , 'password' => md5(trim($request->
getParam('password'))) , 'email' => trim($request->getParam('email')) , 'tel' => trim($request->getParam('tel')) ,
'pc' => $p . '-' . $c , 'regtime' => date('Y-m-d H:i:s') , 'face' => '' , 'pubtimes' => 0 , 'replytimes' => 0 , 'score' => 20 ,
'usertype' => 0);                                             //用户注册信息所构建的数组
    try {
        $this->_userModel->insert($arrayUserInfo);        //保存用户注册信息
    } catch (Zend_Exception $e) {
        $e->getMessage();
    }
    $this->_sessionNamespace->netname = trim($request->getParam('netname'));
    $this->_redirect('/user/register-success');               //定向到用户注册成功提示页面
    exit();
}
```

代码贴士

❶ 为 Zend_Mail 的构造方法指定 utf-8 这个参数的作用是，设定邮件编码为 utf-8 编码。

上述代码保存用户注册信息，主要通过用户表模型 Model_DbTable_User 的 insert()方法实现，该方法的语法格式如下：

```
insert(array insertArray)
```

该方法中主要包含一个关联数组参数，该数组的键名为用户表的字段名，键值为用户输入的注册信息。

为了提高网站形象，提高用户回访数量，在保存用户注册信息前，向注册用户所填写的邮箱中发送一封注册成功提示邮件。发送邮件功能是通过 Zend_Mail 实现的，使用 Zend_Mail 发送邮件相对简单，实例化 Zend_Mail 对象后，只需通过调用该类的几个方法，为邮件指定主题、内容、发件人和收件人等信息，最后再通过 send()方法将邮件发送出去。Zend_Mail 类中常用的方法说明如表 9.8 所示。

表 9.8　Zend_Mail 类中常用的方法说明

属　性　值	说　　明
setSubject()	设置邮件主题
setBody()	设置文本格式的邮件内容
setBodyHtml()	设置 HTML 格式的邮件内容
setFrom()	设置发件人地址
addTo()	设置收件人地址

9.9.4 用户登录的实现过程

用户注册成功后，以后访问本站就可以直接以已注册的账号登录。明日知道网站的用户登录页面如图 9.31 所示。



图 9.31 用户登录页面

下面建立用户登录表单。明日知道网站的登录表单使用 Zend_Form 制作，代码如下：

例程20 代码位置：光盘\TM\09\known\application\forms\User\login.php

```
class Form_User_Login extends Zend_Form                       //定义用户登录表单，继承自 Zend_Form
{
    public function __construct ($options = null)
    {
        parent::__construct($options);                         //调用父类构造函数

        $this->setName('form_login')                           //表单名称
             ->setMethod('post')                               //表单提交方法
             ->setAction($options['baseUrl'] . '/user/login')  //表单提交地址
             ->addAttribs(array(
                 'style' => 'margin:0px; padding:0px'          //表单样式
             ));
❶       $this->addElements(array(                              //为表单增加用户昵称录入文本框
             new Zend_Form_Element_Text('netname', array(
                 'required' => true,                           //指定该表单元素是必要的
                 'label' => '昵称：',                           //元素标签
                 'attribs' => array(                           //元素属性
                     'class' => 'input_login_form',
                     'style' => 'position:absolute; left:32px; top:0px; width:180px; height:18px; line-height:18px'
                 ),
                 'filters' => array('StringTrim'),             //过滤掉首尾空格
                 'validators' => array(
                     array('NotEmpty', true, array('messages' => '・请输入登录昵称'))  //用户昵称非空验证
                 ),
                 'decorators' => array(                        //元素装饰器
                     'ViewHelper',
                     array('HtmlTag', array('tag'=>'dd')),
                     array('Label', array('tag'=>'dt'))
                  )
             )),
             new Zend_Form_Element_Password('password', array(     //为表单增加用户密码录入文本框
                 'required' => true,                               //指定该表单元素是必要的
```

```
            'label' => '密码：',                                         //元素标签
            'attribs' => array(                                           //密码框属性
                'class' => 'input_login_form',
                'style' => 'position:absolute; left:32px; top:0px; width:180px; height:18px; line-height:18px;'
            ),
            'filters' => array('StringTrim'),                             //过滤掉首尾空格
            'validators' => array(                                        //登录密码非空验证
                array('NotEmpty', true, array('messages' => '• 请输入登录密码'))
            ),
            'decorators' => array(                                        //元素装饰器
                'ViewHelper',
                array('HtmlTag', array('tag' => 'dd')),
                array('Label', array('tag' => 'dt'))
             )
        )),
        new Plugin_Form_Element_Vcode('vcode', array(                     //验证码
            'required' => true,                                           //指定该表单元素是必要的
            'label' => '验证码：',                                       //元素标签
            'attribs' => array(                                           //验证码属性
                'textClass' => 'input_login_form',
                'textStyle' =>'position:absolute; left:32px; top:0px; width:60px; height:18px; line-height:18px; ',
                'imageStyle' =>'position:absolute; left:100px; top:0px;',
                'spanStyle' => 'position:absolute; left:42px; top:37px; height:18px',
                'aClass' => 'a4',
                'functionName' => 'changeValidateCode()'
            ),
            'filters' => array('StringTrim'),                             //过滤掉首尾空格
            'validators' => array(
                array('NotEmpty', true, array('messages' => '• 请输入验证码')),    //验证码非空验证
                new Plugin_Validate_VcodeRight()                          //验证码正确性验证
            ),
            'decorators' => array(                                        //元素装饰器
                'ViewHelper',
                array('HtmlTag', array('tag' => 'dd', 'style' => 'height:55px;')),
                array('Label', array('tag' => 'dt'))
             )
        )),
        new Zend_Form_Element_Image('submitImage', array(                 //提交图片按钮
           'required' => false,                                           //该元素值可为空
           'label' => '',
           'src' => $options['baseUrl'] . '/img/btn_login.gif',           //图片地址
           'attribs' => array(
               'style' => 'position:absolute; left:30px; top:0px;'
           )
        ))
    ));
  }
}
```

代码贴士

❶ addElements()方法：用于将表单元素对象添加到表单中。

使用 Zend_Form 制作表单，除可构建表单 UI 界面外，还可以同时实现用户输入内容的过滤、验

证等操作，从而可以大幅度地提高程序开发效率。使用 Zend_Form 构建表单组件的构造方法一般有两个参数，分别为表单元素名称和指定该元素属性的关联数组，该数组常用键值说明如表 9.9 所示。

表 9.9　Zend_Form 表单元素常用键值说明

属　性　值	说　　明
required	用于指定表单元素值是否是必要的
label	表单元素标签
attribs	表单元素属性
filters	对表单元素进行过滤
validators	对表单元素进行校验
decorators	对表单元素 UI 进行装饰

9.10　发表问题模块设计

当用户在明日知道网站搜索问题时，如果没有查找到预期结果，可以将要解决的问题发表到该站，其他用户就可以对问题进行解答或探讨。

9.10.1　发表问题模块概述

如果用户已经注册为本站会员，并成功登录本站，就可以通过明日知道网站的发表问题模块将问题发表出去，其他用户就可以回答该问题，最终可以帮助提问用户获得问题的最佳答案。明日知道网站的发表问题表单主要包括问题主题、问题类别、问题的内容描述、附件、验证码等表单元素，如图 9.32 所示。



图 9.32 发表问题表单

9.10.2　发表问题模块技术分析

发表问题模块中，应该掌握的主要技术包括在线编辑器的制作、图片验证码的制作、文件上传的实现过程。其中，在线编辑器的制作及图片验证码的制作过程已经在前面章节中进行重点讲解，本节介绍通过 Zend Framework 实现文件上传的过程。

（1）用户发表问题表单是通过 Zend_Form 实现的，所以需要使用 Zend_Form_Element_File 类建立文件上传域，代码如下：

例程21　代码位置：光盘\TM\09\known\application\forms\Bbs\Pubtitle.php

```
new Zend_Form_Element_File('file1', array(                //文件选择域
    'required' => false,                                  //指定文件上传域并非必要
    'label' => '附件：',                                   //标签
    'maxFileSize' => '30000000',                          //上传文件大小
    'description' => '请选择要上传的附件',                   //描述
```



```
        'attribs' => array(                                     //文件上传域属性
            'size' => '60',
            'class' => 'input_pubtitle_form'
        ),
        'validators' => array(                                  //非空验证
            array('NotEmpty', false, array('messages' => '请输入内容'))
        ),
        'decorators' => array(                                  //装饰器
            'File',
            array('Description', array('tag' => 'dt')),
            array('HtmlTag', array('tag' => 'dd') ),
            array('Label', array('tag' => 'dt'))
        )
)),
```

在讲解用户登录表单时，已经详细介绍 Zend_Form 表单及表单元素常用属性的含义，上述代码中，使用“'maxFileSize' => '30000000'”指定所允许上传附件的最大值，运行发表问题模块，上述代码将转换为如下 HTML 代码：

```
<input type="hidden" name="MAX_FILE_SIZE" value="30000000" id="MAX_FILE_SIZE">
<input type="file" name="file1" id="file1" size="60" class="input_pubtitle_form">
```

（2）当用户提交发表问题表单后，用户提交的内容将被提交到 QuestionController 控制器的 pubtitleAction 动作进行处理，其中用户实现文件上传的关键代码如下：

例程22　代码位置：光盘\TM\09\known\application\modules\default\controllers\QuestionController.php

```
$adapter = new Zend_File_Transfer_Adapter_Http();           //构建上传适配器
$upfileName = '';                                           //上传后，保存在服务器中的文件名
if ($adapter->getFileName('file1') != null) {               //如果用户已经选择了上传文件
    $upfileDir = $this->_config['bbs']['upfilesdir'];       //从配置文件中读取上传文件保存目录
    if (! is_dir($upfileDir)) {                             //判断上传目录是否存在，如果不存在则创建该目录
        mkdir($upfileDir);
    }
    $adapter->setDestination($upfileDir);                   //为上传适配器设置上传目录
$arrayOldFileName = array_reverse(explode('.', basename($adapter->getFileName('file1'))));
//提取上传文件原来的名字
    $extendsFileName = '.' . $arrayOldFileName[0];          //获取上传文件扩展名
    $upfileName = date('YmdHis') . mt_rand(1000, 9999) . $extendsFileName;   //为上传文件指定新名称，用时
将和一个 4 位随机数组成
    $upfilePathAndName = $upfileDir . '/' . $upfileName;  //文件在服务器中保存目录及名称
    $adapter->addFilter('Rename', array('target' => $upfilePathAndName , 'overwrite' => true), 'file1');//更改上传
文件名
    $adapter->addValidator('Size', false, array('min' => '0kB' , 'max' => '2MB' , 'bytestring' => false , 'messages'
=> '• 您所上传的文件不能超过 2MB'));                       //指定上传文件的范围
    $adapter->addValidator('ExcludeExtension', false, array('php' , 'exe' , 'messages' => '• 您上传的文件类型不
允许'));                                                    //指定上传文件扩展名
}
```

上述代码实现文件上传是通过 Zend_File 实现的，由于本模块上传方式采用 HTTP 方式，所以实现文件上传主要通过 Zend_File_Transfer_Adapter_Http 这个适配器实现，该适配器常用的方法说明如表 9.10

所示。

表 9.10　Zend_File_Transfer_Adapter_Http 适配器常用的方法说明

属　性　值	说　　明
getFileName()	获取上传文件的名称
setDestination()	指定上传文件保存到服务器中的目录
addFilter()	为上传增加过滤规则
addValidator()	为上传增加校验规则

9.10.3　发表问题模块实现过程

制作明日知道网站发表问题模块，主要包括发表问题表单视图的建立和保存问题内容两个关键环节。

（1）明日知道网站发表问题表单使用 Zend_Form 实现，通过 Zend_Form 实现表单的制作可以简化对提交数据的过滤、校验、UI 布局的操作，从而提高开发效率，便于开发人员分层开发及日后生产环境下的维护工作。本站实现发表问题表单的代码如下：

例程23　代码位置：光盘\TM\09\known\application\forms\Bbs\Pubtitle.php

```
class Form_Bbs_Pubtitle extends Zend_Form
{
    public function __construct ($options = null)
    {
        parent::__construct(null);                                  //调用父类构造方法
        $this->setName('form_pubtitle')                             //表单名称
             ->setMethod('post')                                    //表单提交方法
             ->setAction($options['baseUrl'].'/question/pubtitle')  //表单提交地址
             ->addAttribs(array(                                    //表单属性
                 'enctype' => 'multipart/form-data',
                 'style' => 'margin:0px; padding:0px;',
             ));
        $this->addElements(array(
            new Zend_Form_Element_Text('title', array(
                …                                                   //省略代码用于设置问题主题元素的属性
            )),
            new Zend_Form_Element_Select('bbstypeid', array(
                …                                                   //省略代码用于设置问题类别下拉列表属性
            )),
            new Plugin_Form_Element_Editor('content', array(
                …                                                   //省略代码用于设置问题内容编辑器的属性
             )),
            new Zend_Form_Element_File('file1', array(              //文件选择域
                …                                                   //省略代码用于设置文件选择框的属性
            )),
            new Plugin_Form_Element_Vcode('vcode', array(
                …                                                   //省略代码用于设置验证码的属性
            )),
            new Zend_Form_Element_Button('submitButton', array(
```

```
            ...                                          //省略代码用于设置提交按钮的属性
        )),
        new Zend_Form_Element_Hidden('titleid', array(  //在编辑问题时，该隐藏域用于保存问题 ID 的值
            'value' => $options['titleid']
        ))
    ));
  }
}
```

上述 Form_Bbs_Pubtitle 类继承自 Zend_Form 类，这样该类就可以使用 Zend_Form 类的所有方法，如使用 setName()方法设置表单名称、setMethod()方法设置表单提交方法、setAction()方法设置表单提交地址等，其中 Zend_Form 表单元素常用类说明如表 9.11 所示。

表 9.11　Zend_Form 表单元素常用类说明

属　性　值	说　　明
Zend_Form_Element_Text	生成文本框
Zend_Form_Element_Select	生成下拉列表框
Zend_Form_Element_File	生成文件选择域
Zend_Form_Element_Button	生成按钮
Zend_Form_Element_Hidden	生成隐藏域
Zend_Form_Element_Checkbox	生成复选框
Zend_Form_Element_Image	生成图片按钮
Zend_Form_Element_Password	生成密码框
Zend_Form_Element_Radio	生成单选按钮
Zend_Form_Element_Reset	生成"重置"按钮
Zend_Form_Element_Submit	生成"提交"按钮
Zend_Form_Element_Textarea	生成文本域
Zend_Form_Element_Text	生成文本框

（2）用户提交表单后，输入的内容将被提交到 QuestionController 控制器的 pubtitleAction 动作并保存。保存用户提交内容的核心代码如下：

例程24　代码位置：光盘\TM\09\known\application\modules\default\controllers\QuestionController.php

```
$pubUser = $this->_userModel->findByNetname($this->_auth->getIdentity());  //获取发帖用户信息
$arrayInsert = array('title' => $formData['title'] , 'content' => $formData['content'] , 'unhtmlcontent' => $this->view->
unHtml($formData['content']) ,
'addtime' => date('Y-m-d H:i:s') , 'istop' => false , 'user_id' => $pubUser['id'] , 'bbstype_id' => $formData ['bbstypeid'] ,
'browse' => 0 , 'filename' =>
$upfileName , 'isjh' => false);                          //发帖内容数组
$this->_titleModel->insert($arrayInsert);                //使用问题模型的 insert()方法，将问题信息保存到数据库
$lastTitle = $this->_titleModel->findLastByUserid($pubUser['id']);      //查找该用户最后发帖信息，即刚插入帖子
的信息
$this->_redirect('question/thread/param/' . $lastTitle['id']);          //使页面定位到该帖的详细信息页
```

上述代码首先通过用户模型的 findByNetname()方法获得当前登录用户的信息，然后将问题信息保存到数据库的 tb_title 表中，成功保存问题信息后，将页面定向到所发问题的详细信息页。

9.11 回复问题模块设计

用户成功注册为明日知道网站会员后，不仅可以发表问题，还可以回复问题，与其他用户进行讨论。

9.11.1 回复问题模块概述

明日知道网站的问题回复表单采用 Zend_Form 制作，主要包括回复主题、回复内容和验证码等表单元素，如图 9.33 所示。



图 9.33　问题回复表单

9.11.2 回复问题模块技术分析

回复问题模块中，应该掌握的主要技术包括在线编辑器的制作和使用方法、图片验证码的制作方法，以及问题引用的实现，其中在线编辑器和图片验证的实现方法在上面已经介绍，下面介绍明日知道网站问题引用的实现过程，关键代码如下：

```
$value = '<FIELDSET><LEGEND>引自：楼主</LEGEND>引用的内容</FIELDSET>';
```

将上述代码作为编辑器的 value 值，将在编辑器显示如图 9.34 所示的效果。通过上述代码可知，引用可以通过 HTML 标签的<FIELDSET>标签实现，<FILELDSET>标签用于在文本或其他元素外绘制一个边框，如果在该标签内使用<LEGEND>子标签指定标题，则在边框左上侧还会显示指定的标题。

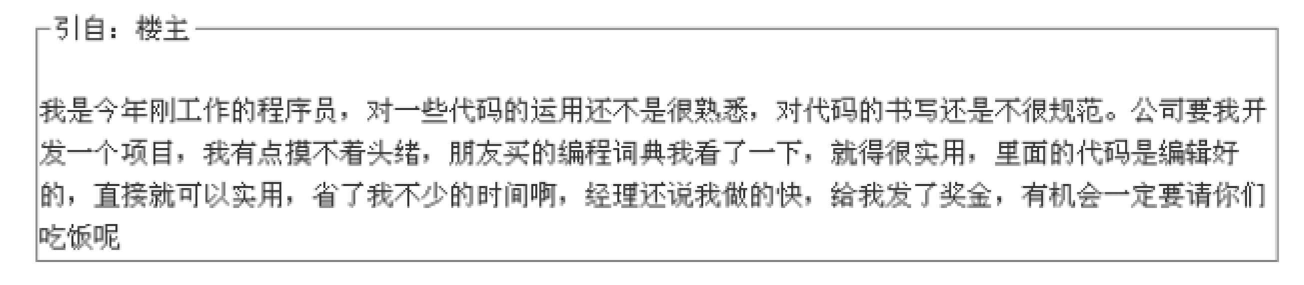
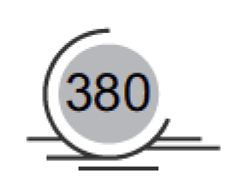


图 9.34　问题引用



9.11.3　回复问题模块实现过程

问题回复模块，主要包括回复表单的设计和保存回复信息两个主要过程。

（1）明日知道网站的问题回复表单采用 Zend_Form 制作，关键代码如下：

例程25　代码位置：光盘\TM\09\known\application\modules\default\controllers\QuestionController.php

```
class Form_Bbs_Reply extends Zend_Form
{
    public function __construct ($options = null)
    {
        parent::__construct(null);                            //调用父类构造函数

        $this->setName('form_reply')                          //设置回复表单名称
             ->setMethod('post')                              //设置表单提交方法
->setAction($options['baseUrl'].'/question/thread/param/' . $options['titleid'] . '-1-T')
//设置表单提交地址
             ->addAttribs(array(                              //设置提交表单属性
                 'style' => 'margin:0px; padding:0px;',
        ));
        $this->addElements(array(
            new Zend_Form_Element_Text('title', array(
            ......                                            //省略代码用于设置回复主题文本框属性
            )),
            new Plugin_Form_Element_Editor('content', array(
            ...                                               //省略代码用于设置编辑器属性
            )),
            new Plugin_Form_Element_Vcode('vcode', array(
            ...                                               //省略代码用于设置验证码属性
            )),
            new Zend_Form_Element_Button('submitButton', array(
            ...                                               //省略代码用于设置“提交”按钮属性
            ))
        ));
    }
}
```

上述代码中的 Form_Bbs_Reply 类继承自 Zend_Form 类，通过 Zend_Form 类中定义的方法来为回复问题表单设置属性，并通过 addElements()方法将表单元素对象添加到问题回复表单中。

（2）用户输入完成回复内容，提交表单后，所有回复内容将被提交到 QuestionController 控制器的 threadAction 动作中进行保存，其中用于保存用户回复信息的关键代码如下：

例程26　代码位置：光盘\TM\09\known\application\modules\default\controllers\QuestionController.php

```
if ($this->_request->isPost()) {                              //判断用户是否提交了表单
    $formData = $this->_request->getPost();                   //获取表单提交数据所组成的数组
    if ($replyForm->isValid($formData)) {                     //如果通过表单验证
        $replyid = $this->_request->getParam('replyid');      //获得问题类别 ID
        if (isset($replyid) && $replyid != null) {            //如果设置了问题类别 ID，则说明要进行更改问题回复
操作，否则进行添加问题回复操作
$updateArray = array('title' => $formData['title'] , 'content' => $formData['content']);
```

```
//更改问题回复数组
            $where = $this->_replyModel->getAdapter()->quoteInto('id = ?', $replyid);    //更改条件
            $this->_replyModel->update($updateArray, $where);                    //更改数据
            $this->_redirect('/question/thread/param/' . $titleid . '-' . $this->_request->getParam('topage') . '#r' .
$replyid);                                                           //重定向到问题详细信息页面
        } else {
            $replyUser = $this->_userModel->findByNetname($this->_auth->getIdentity());   //获取回复人信息
            $insertArray = array('title' => $formData['title'] , 'content' => $formData['content'] , 'addtime' => date
('Y-m-d H:i:s') , 'user_id' => $replyUser['id'] , 'title_id' => $arrayTitle['tid'] , 'topindex' => 0 , 'bbstype_id' => $arrayTitle
['bbstype_id']);                                                     //添加问题回复数组
            $this->_replyModel->insert($insertArray);                            //添加数据
            $totalReply = $arrayTitle['totalreply'] + 1;
❶           $totalPage = ceil($totalReply / $pageSize);
❷           $this->_redirect('/question/thread/param/' . $titleid . '-' . $totalPage . '#b');   //重定向到问题详细信息页面
        }
    } else {
        foreach ($replyForm->getMessages() as $messageArray) {               //获取错误信息
            foreach ($messageArray as $message) {
                $errorMsg .= $message . '<br />';
            }
        }
        $replyForm->populate($formData);                                     //表单数据回填
    }
}
```

代码贴士

❶ 函数 ceil()的作用是对浮点数进行向上取整。

❷ _redirect()方法的作用是使当前页面重定向到其他页面。

上述代码首先通过当前控制器对象的_request 属性的 isPost()方法，判断用户是否提交表单，如果是则首先对表单提交的数据进行数据校验，如果成功通过校验，则通过回复表（tb_reply）模型的 insert()方法，将回复信息保存到数据库中。

9.12 程序调试与错误处理

在程序开发过程中，难免会遇到各种错误，为了能够快速地解决程序开发过程中所遇到的错误，需要开发人员具有良好的的编程习惯和敏锐的观察能力，并不断地培养个人的开发经验。本节将对程序开发和运行过程中常见错误的调试和处理方法进行讲解。

9.12.1 项目开发过程中常见错误及解决方法

为了能够显示项目开发过程中的错误提示，需要开发php.ini文件的display_startup_errors和display_errors 项，或者在本站的 application.ini 文件中加入如下配置来使项目在开发过程中显示错误提示。

```
phpSettings.display_startup_errors = 1
phpSettings.display_errors = 1
```



这里需要注意，在项目投入生产后，需要将上述两项的值设为 0，这样在运行过程中即使出现错误，也不会在页面中打印出错误信息，从而可以防止网站的漏洞被非法用户发现，有效地提高网站的安全性。开启上述错误提示信息后，在项目开发过程中，一旦遇到错误，开发人员可以根据错误提示推断错误所在并予以解决。

在 PHP 5 以上版本中，引入了异常处理机制，开发人员通过异常处理可以捕获程序运行时的错误。异常处理中主要有 try、catch 和 throw 这 3 个关键字，其语法结构如下：

```
try{
    //要执行的代码块
}catch(Exception $e){
    //异常处理方式，这里可以打印错误信息或使用 throw 语句抛出异常
}
```

用户要执行的代码块编写在 try/catch 代码块之间，通过 catch 语句指定要捕获的异常，并通过 throw 语句抛出异常。

9.12.2　自定义错误页面

为了提高页面的友好程度，隐藏项目生产环境下的错误，经常会在项目中制作一个单独的用于显示错误提示的页面。明日知道网站的错误提示页面如图 9.35 所示。



图 9.35　自定义的错误提示页面

使用 Zend Framework 框架制作项目时，有一个专门的错误控制器 ErrorController 用来显示页面错误提示，开发人员只需在该控制器中定义一个名为 errorAction 的动作即可。明日知道网站中的错误控制器代码如下：

例程27　代码位置：光盘\TM\09\known\application\modules\default\controllers\QuestionController.php

```
class ErrorController extends Zend_Controller_Action
{
    public function errorAction ()
    {
        $this->view->title = "明日知道网站错误提示";           //页面标题
        $this->_helper->layout->disableLayout();               //去掉布局
    }
}
```

定义完成上述错误控制器后，还需要定义 erro.phtml 视图，该视图用来显示错误提示的内容。

9.13　本 章 总 结

明日知道网站力求打造成为国内知名的专业软件开发技术搜索和交流平台，通过该平台开发人员

和编程初学者可以查找到大部分开发和学习过程中可能会遇到的技术难题，即使没有查找到预期结果，还可以通过该平台将问题发表出去，这样就可以与其他网友探讨，最终将问题解决。

在开发过程中，通过严谨的需求分析、功能分析和数据库分析的全过程，并充分考虑到生产环境下的升级与维护等后期工作，本站采用 Zend 公司研发的 MVC 框架 Zend Framework 进行开发。在开发过程中，使用到目前 Web 开发领域热门技术，如在线编辑器、图片验证码、类似 Google 搜索条的下拉关键字提示列表、关键字分词查询和查询结果中关键字描红输出等。

通过学习本章内容，可以对项目的完整开发流程有所了解，并理解 MVC 开发模式的含义。技术方面，不仅可以掌握目前 Web 应用的主流模块和功能的制作方法，而且可以熟练应用 Zend Framework 框架进行项目开发。如果能够潜心学习本站内容，相信定能受益匪浅，使个人技术水平有显著的提高。

第10章

电子商务网站

（Apache+PHP+ADODB+Smarty+Ajax+MySQL 实现）

随着 20 世纪 PC 机（个人计算机）的发展和互联网的普及，电子商务从报文时代进入到了 Internet 时代，并逐渐被大众所了解和接受。电子商务（Electronic Commerce，EC）是目前发展较快的一种商务模式。迄今为止，不同领域的人对 EC 的理解各有不同。简单地说，EC 是一种基于 Internet，利用计算机硬件、软件等现有设备和协议进行各种商务活动的方式。

通过阅读本章，可以学习到：

- 了解如何进行系统分析
- 了解数据库设计流程
- 熟悉搭建系统架构的方法
- 掌握 Smarty 模板技术的应用
- 掌握 ADODB 类库技术的应用
- 熟悉 Smarty 模板和 ADODB 类库的配置方法
- 掌握注册即时验证的实现方法
- 掌握简单的树形菜单的实现方法
- 掌握购物车的实现方法
- 掌握订单的处理方法

10.1 开发背景

自 20 世纪 90 年代，互联网的蓬勃发展，为企业提供了一个全新的机遇。企业网站、电子商务成为热门话题。其中，电子商务更是关系到经济结构、产业升级和国家整体经济竞争力。为此，我国已经将发展电子商务列为信息化建设的重要内容，并努力创造条件，积极地推进电子商务的发展。

据美国在线（AOL）和 Henley Centre 联合进行的一项调查显示：国外有 80%的受调查者会选择网上购物或寻求帮助，10%的受调查者会选择熟悉的品牌或厂商来购买。而在国内，自 1997 年拉开了电子商务的序幕，短短的 10 年时间里，全国已有 4 万家商业网站，几乎每天都有新的网站诞生，厂商所在地也从上海、广州、深圳等沿海发达地区扩展到全国各大、中城市。

10.2 需求分析

随着“地球村”概念的兴起，网络已经深入到人们生活的每一个角落。世界越来越小，信息的传播越来越快，内容也越来越丰富。现在，人们对于在网络上寻求信息和服务已不再满足于简单的信息获取上，人们更多的是需要在网上实现方便的、便捷的、可交互式的网络服务。电子商务则正好满足了人们的需求。它可以让人们在网上实现互动的交流及足不出户地购买产品，向企业发表自己的意见、服务需求及有关投诉，并且通过网站的交互式操作向企业进行产品的咨询、得到相应的回馈及技术支持。精明的商家绝不会错过这样庞大的市场，越来越多的企业已经开展了电子商务活动。加入电子商务的行列也许不会让企业马上见到效益，但不加入则一定会被时代所抛弃。

10.3 系统分析

10.3.1 系统目标

根据客户提供的需求和对实际情况的考察与分析，该电子商务应该具备如下特点：

☑ 首页设计要能够吸引用户的目光，整个页面要以简洁为主，突出重点。
☑ 可操作性强，避免复杂的、有异议的链接。
☑ 浏览速度快，尽量避免长时间打不开页面的情况发生。
☑ 商品信息部分有实物图例，图像清楚、文字醒目。
☑ 详细的商品查询功能，可以通过商品的各个属性来搜索。
☑ 详细的流程介绍，从浏览商品到购买结账，各个步骤之间的联系最好能以图例来说明。
☑ 提供在线咨询。
☑ 后台可以对用户信息和商品信息进行详尽的查看和管理。
☑ 订单管理。
☑ 易维护，并提供二次开发支持。

10.3.2　系统功能结构

电子商务平台分前台系统和后台系统。下面分别给出前、后台的系统功能结构图。电子商务前台系统功能结构如图 10.1 所示。

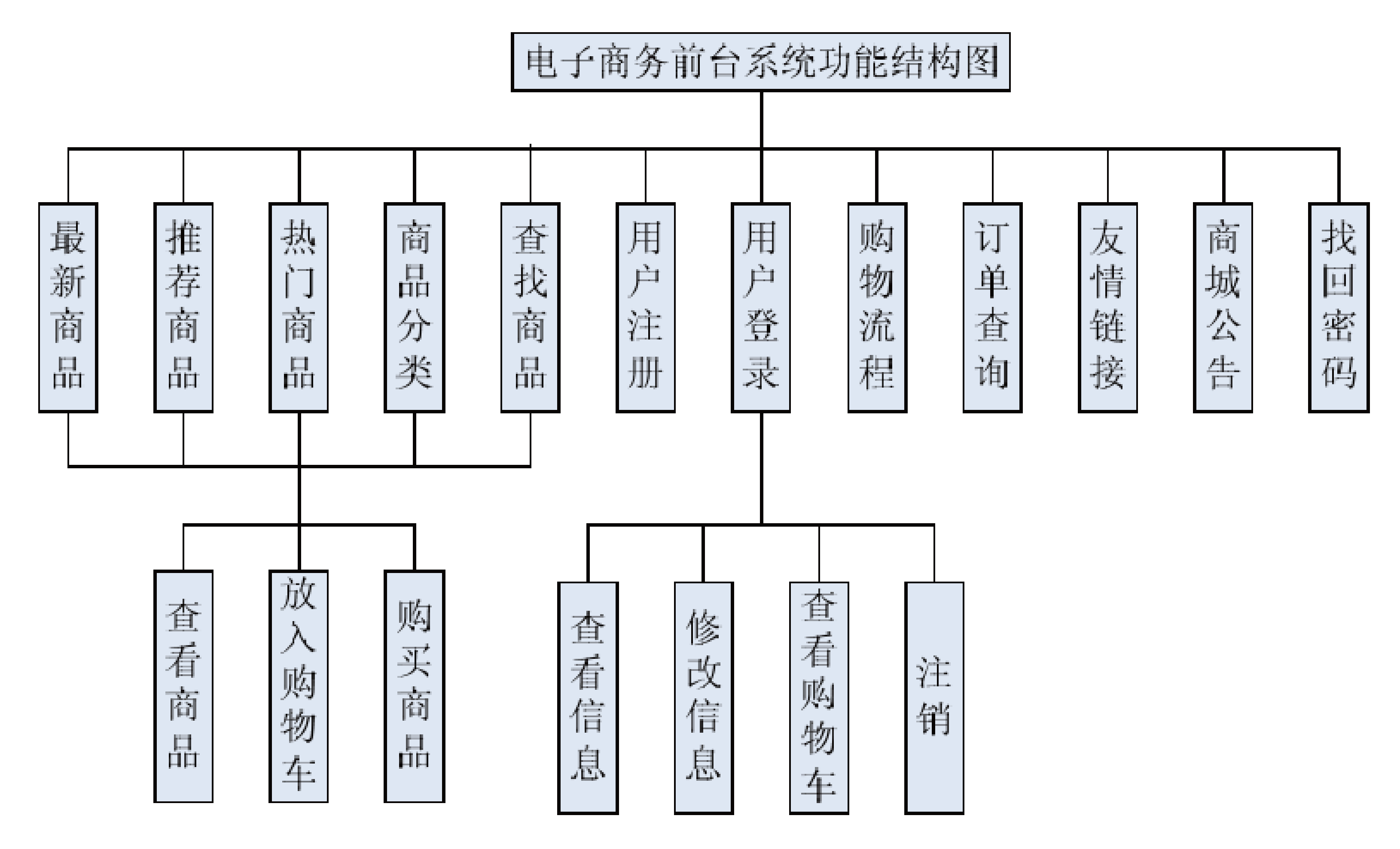


图 10.1　电子商务前台系统功能结构图

电子商务后台系统功能结构图如图 10.2 所示。

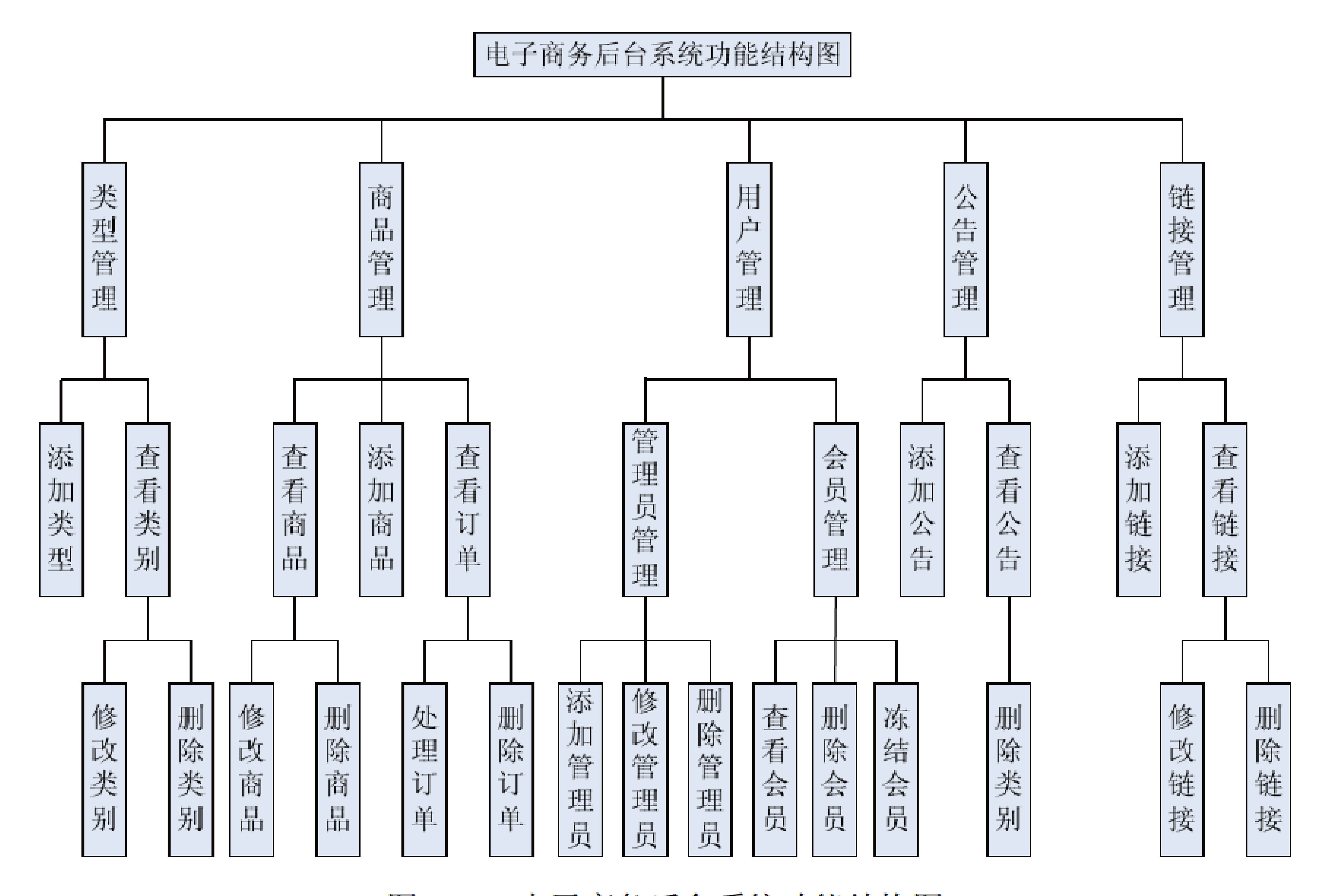
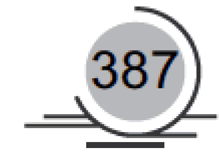


图 10.2　电子商务后台系统功能结构图

10.3.3　功能预览

电子商务网站由多个功能模块组成，为了让读者对本系统有个初步的了解和认识，下面列出几个



典型功能的页面，其他页面参见光盘中的源程序。

鹏逊购物系统主页如图 10.3 所示，该页面展示网站的部分最新商品、热门商品、推荐商品以及网站的最新公告和会员登录窗口。最新商品展示页面如图 10.4 所示，该页面分页展示网站的所有最新商品。



图 10.3　网站主页面



图 10.4　最新商品展示

购物车页面如图 10.5 所示，该页面展示会员在本站购买的商品。

收银台页面如图 10.6 所示，该页面完成收货人信息的填写，并提交订单。

我的购物车

	商品名称	购买数量	市场价格	会员价格	折扣率	合计
☑	自行车	1	450	405	9	405
☑	家庭影院	2	4500	4050	9	8100
☐	洗衣机	3	1000	900	9	2700
☐	咖啡杯	4	20	18	9	72

全选 反选　删除选择　继续购物　去收银台　共计：11277 元

图 10.5　购物车



订单信息

收货人姓名：	明日科技	邮编：	130025
联系电话：	0431-84972266		
地址：	长春市		
送货方式：	快递		
付款方式：	支付宝		

提交订单

图 10.6　收银台

10.3.4　系统流程图

为了便于开发人员了解系统各个功能模块之间的联系及完整的购物流程，下面给出系统的流程图，如图 10.7 所示。

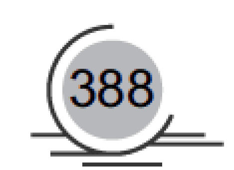
10.3.5　开发环境

在开发电子商务平台时，该项目使用的软件开发环境如下。

1．服务器端

☑　操作系统：Windows Server 2003/Linux（推荐）。

☑　服务器：Apache 2.2.8。





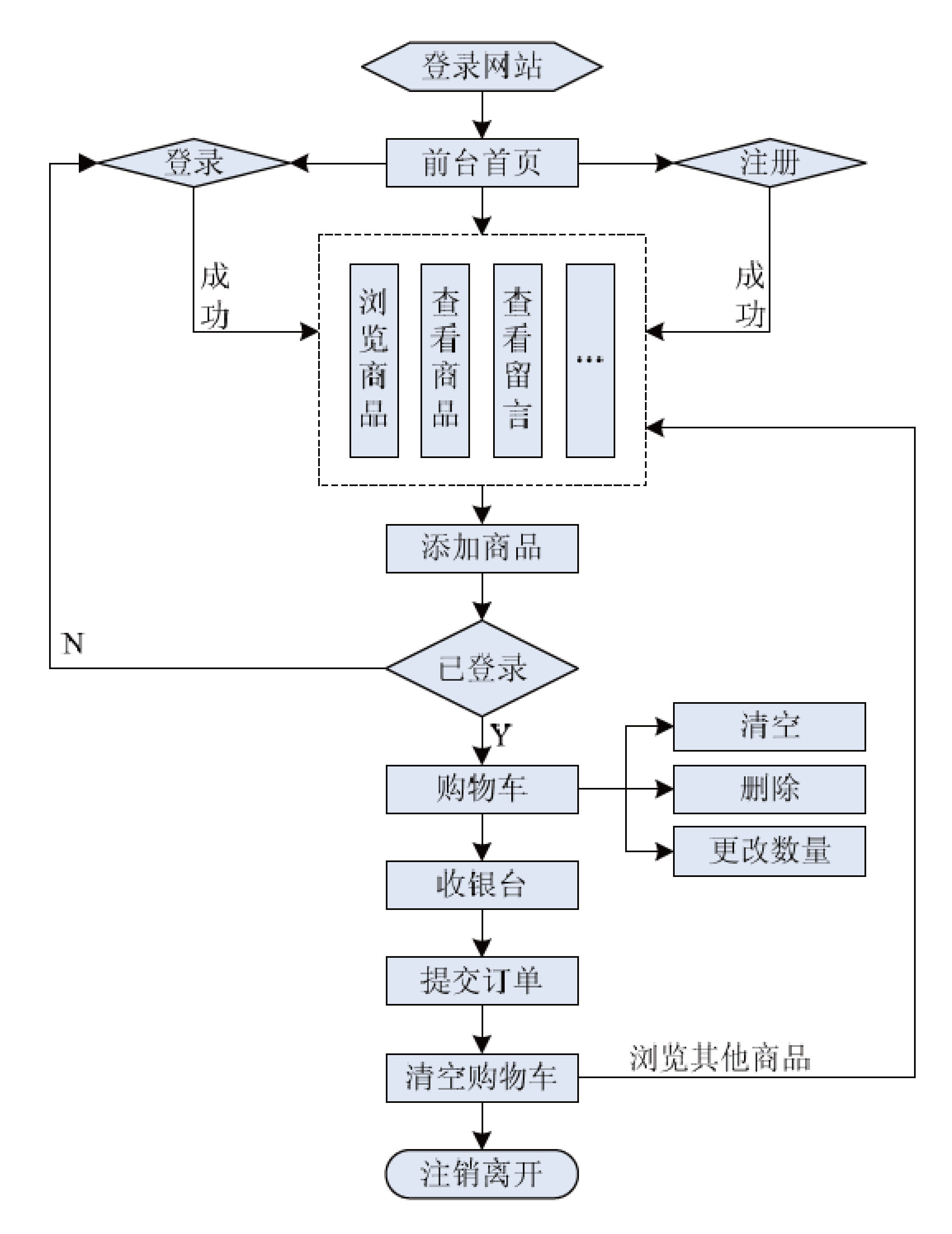


图 10.7　系统流程图

- ☑ PHP 软件：PHP 5.2.6。
- ☑ 数据库：MySQL 5.0.51。
- ☑ MySQL 图形化管理软件：phpMyAdmin-2.10.3。
- ☑ ADODB 类库。
- ☑ Smarty 模板。
- ☑ 开发工具：Dreamweaver 8。
- ☑ 浏览器：IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

2．客户端

- ☑ 浏览器：推荐 IE 6.0 及以上版本。
- ☑ 分辨率：最佳效果为 1024×768 像素。

10.3.6　文件夹组织结构

编写代码之前，可以把系统中可能用到的文件夹先创建出来（例如，创建一个名为 images 的文件夹，用于保存程序中所使用的图片），这样不但可以方便以后的开发工作，也可以规范系统的整体架构。因为本项目使用的是 Smarty+ADODB 技术，所以目录较多。下面介绍一下本系统的目录结构（到三级目录），如图 10.8 所示。

```
12\ShoppingSystem —— 电子商城系统目录
  admin —— 系统后台目录
    css —— css样式文件
    func —— 自定义函数文件
    images —— 页面背景图片文件
    inc —— 公共资源文件
    js —— javascript脚本文件
    system —— ADODB类库、Smarty类库的操作文件夹
      adodb —— adodb类库
      cache —— Smarty缓存文件
      configs —— Smarty配置文件
      smarty —— Smarty模板类库
      templates_c —— Smarty临时文件
      system.class.inc.php —— 数据库连接、管理和分页类文件
      system.inc.php —— 数据库连接、管理、分页类和Smarty配置类的实例化文件
      system.smarty.inc.php —— Smarty模板配置类文件
  css —— css样式文件
  images —— 页面背景图片文件
  inc —— 公共资源文件
  js —— javascript脚本文件
  pics —— 商品图片文件
  system —— ADODB类库、Smarty类库的操作文件夹
    adodb —— adodb类库
    cache —— Smarty缓存文件
    configs —— Smarty配置文件
    smarty —— Smarty模板类库
    templates_c —— Smarty临时文件
    system.class.inc.php —— 数据库连接、管理和分页类文件
    system.inc.php —— 数据库连接、管理、分页类和Smarty配置类的实例化文件
    system.smarty.inc.php —— Smarty模板配置类文件
```

图 10.8　电子商务网站文件夹组织结构图

10.4　数据库设计

无论是什么系统软件，其最根本的功能就是对数据的操作与使用。所以，一定要先做好数据的分析、设计与实现，然后再实现对应的功能模块。

10.4.1　数据库分析

根据需求分析和系统的功能流程图，找出需要保存的信息数据（也可以理解为现实世界中的实体），并将其转化为原始数据（属性类型）形式。这种描述现实世界的概念模型，可以使用 E-R 图来表示，也就是实体-联系图。最后将 E-R 图转换为关系数据库。这里重点介绍几个 E-R 图。

1．会员信息实体

会员信息实体包括编号、名称、密码、E-mail、身份证号、固定电话、QQ、密码保护、密码答案、邮编、注册日期、真实姓名等属性。会员信息实体 E-R 图如图 10.9 所示。

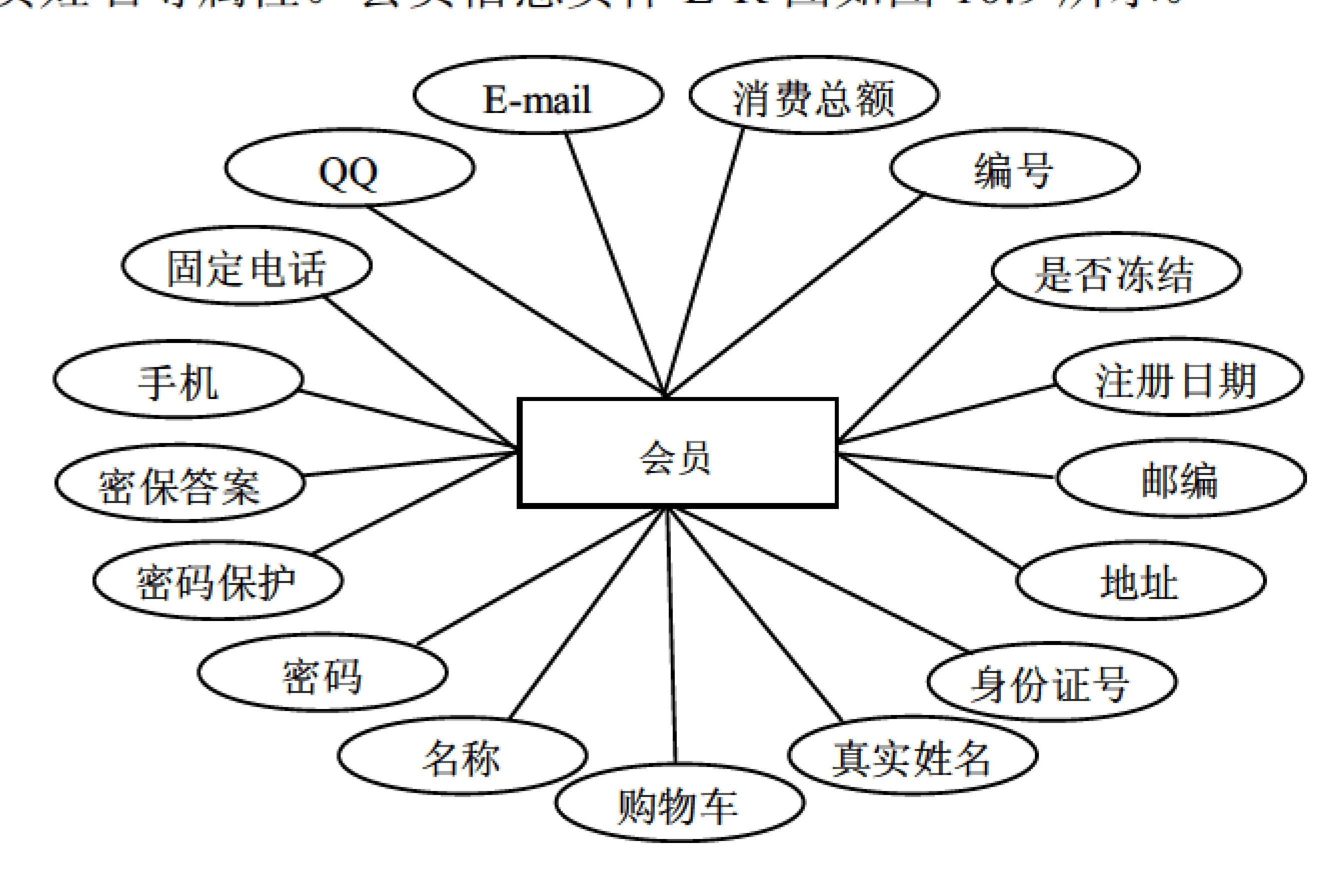


图 10.9　会员信息实体 E-R 图

2．商品信息实体

商品信息实体包括编号、名称、介绍、添加日期、型号、图片、库存、销售量、类型、会员价格、市场价格、是否新品等属性。商品信息实体 E-R 图如图 10.10 所示。

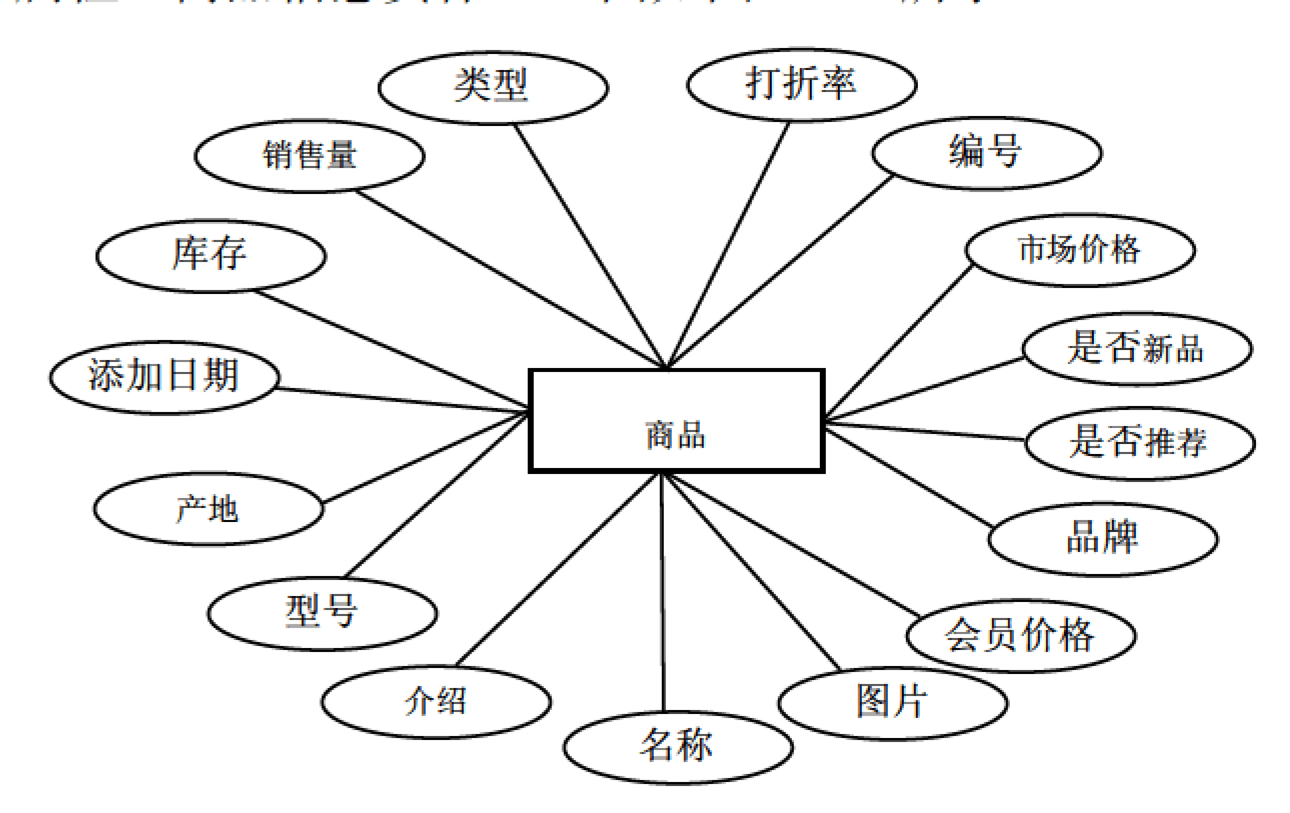


图 10.10　商品信息实体 E-R 图

除上面介绍的 2 个 E-R 图，还有商品订单实体、商品评价实体、公告实体、管理员实体、类型实体和友情链接实体等，限于篇幅，这里仅列出主要的实体 E-R 图。

10.4.2　创建数据库和数据表

系统 E-R 图设计完成后，接下来根据 E-R 图来创建数据库和数据表。首先来看一下电子商务平台所使用的数据表情况，如图 10.11 所示。


服务器: localhost ▸ 数据库: db_shop

表	类型	整理	说明
tb_admin	MyISAM	gb2312_chinese_ci	管理员信息表
tb_class	MyISAM	gb2312_chinese_ci	商品类型表
tb_commo	MyISAM	gb2312_chinese_ci	商品信息表
tb_form	MyISAM	gb2312_chinese_ci	商品订单表
tb_links	MyISAM	gb2312_chinese_ci	友情链接表
tb_opinion	MyISAM	gb2312_chinese_ci	商品评论表
tb_public	MyISAM	gb2312_chinese_ci	公告信息表
tb_user	MyISAM	gb2312_chinese_ci	会员信息表

图 10.11　电子商务数据表

下面来看各个数据表的结构和字段说明。

☑　tb_admin（管理员信息表）

管理员信息表主要用于存储管理员的信息，其结构如图 10.12 所示。

☑　tb_class（商品类型表）

商品类型表主要用于添加商品的类别，可以设定多个子类别（目前最多只能到二级子类别），其结构如图 10.13 所示。

服务器: localhost ▸ 数据库: db_shop ▸ 表 : tb_admin

字段	类型	整理	属性	Null	默认	额外	说明
id	int(4)			否		auto_increment	自动编号
name	varchar(50)	gb2312_chinese_ci		否			管理员账号
pwd	varchar(20)	gb2312_chinese_ci		否			管理员密码

图 10.12　管理员信息表结构

服务器: localhost ▸ 数据库: db_shop ▸ 表 : tb_class

字段	类型	整理	属性	Null	默认	额外	说明
id	int(4)			否		auto_increment	自动编号
name	varchar(20)	gb2312_chinese_ci		否			类型名称
level	int(1)			否	1		等级

图 10.13　商品类型表结构

☑　tb_commo（商品信息表）

商品信息表主要用于存储关于商品的相关信息，其结构如图 10.14 所示。

此外还有商品订单表、商品评论表、公告信息表、友情链接表和会员信息表，限于篇幅，这里不再介绍，读者可参见本书附赠光盘中的数据库文件。

服务器: localhost ▸ 数据库: db_shop ▸ 表 : tb_commo

字段	类型	整理	属性	Null	默认	额外	说明
id	int(4)			否		auto_increment	自动编号
name	varchar(50)	gb2312_chinese_ci		否			商品名称
pics	varchar(200)	gb2312_chinese_ci		否			商品图片
info	mediumtext	gb2312_chinese_ci		否			商品介绍
addtime	date			否			添加时间
area	varchar(50)	gb2312_chinese_ci		否			商品产地
model	varchar(50)	gb2312_chinese_ci		否			商品型号
class	varchar(50)	gb2312_chinese_ci		否			商品类别
brand	varchar(50)	gb2312_chinese_ci		否			商品品牌
stocks	int(4)			否	1		商品库存
sell	int(4)			否	0		商品销量
m_price	float			否			市场价格
v_price	float			否			会员价格
fold	int(2)			否	9		打折率
isnew	int(1)			否	1		是否新品
isnom	int(1)			否	0		是否推荐

图 10.14　商品信息表结构

10.5　公共文件设计

公共模块就是将多个页面都可能使用到的代码写成单独的文件，在使用时只要用 include 或 require 语句将文件包含进来即可，如本系统中的数据库连接、管理和分页类文件，Smarty 模板配置类文件，类的实例化文件，CSS 样式表文件，js 脚本文件等。以前台系统为例，下面给出主要的公共文件，后台的公共文件与前台大同小异。

10.5.1　数据库连接、管理和分页类文件

在数据库连接、管理和分页类文件中，定义 3 个类，分别是 ConDB 数据库连接类，实现通过 ADODB 连接 MySQL 数据库；AdminDB 数据库管理类，使用 ADODB 类库中的方法执行对数据库中数据的查询、添加、更新和删除操作；SepPage 分页类，用于对商城中的数据进行分页输出。

例程01　代码位置：光盘\TM\10\ShoppingSystem\system\system.class.inc.php

```
<?php
//数据库连接类
class ConnDB{
	var $dbtype;
	var $host;
	var $user;
	var $pwd;
	var $dbname;
	var $debug;
	var $conn;
	function ConnDB($dbtype,$host,$user,$pwd,$dbname,$debug=false){		//构造方法，为成员变量赋值
		$this->dbtype=$dbtype;
		$this->host=$host;
		$this->user=$user;
		$this->pwd=$pwd;
		$this->dbname=$dbname;
		$this->debug=$debug;
```

```
    }
    function GetConnId(){                              //实现与不同数据库的连接并返回连接对象
     require("adodb/adodb.inc.php");                   //调用 ADODB 类库文件
        if($this->dbtype=="mysql"){                    //判断成员变量传递的数据库类型
                $this->conn=NewADOConnection("mysql");     //执行与 MySQL 数据库的连接
                $this->conn->Connect($this->host,$this->user,$this->pwd,$this->dbname); //用户、密码
        }
                $this->conn->Execute("set names gb2312");    //设置数据库的编码格式
        if($this->dbtype=="mysql")
                $this->conn->debug=$this->debug;
        return $this->conn;                            //返回连接对象
    }
     function CloseConnId(){                           //定义关闭数据库的方法
                $this->conn->Disconnect();             //执行关闭的操作
    }
}
//数据库管理类
class AdminDB{
     function ExecSQL($sqlstr,$conn){                  //定义方法，参数为 SQL 语句和连接数据库返回的对象
        $sqltype=strtolower(substr(trim($sqlstr),0,6)); //截取 SQL 中的前 6 个字符串，并转换成小写
        $rs=$conn->Execute($sqlstr);                   //执行 SQL 语句
        if($sqltype=="select"){                        //判断如果 SQL 语句的类型为 SELECT
            $array=$rs->GetRows();                     //执行该语句，获取查询结果
            if(count($array)==0 || $rs==false)         //判断语句是否执行成功
                return false;                          //如果查询结果为 0，或者执行失败，则返回 false
            else
                return $array;                         //否则返回查询结果的数组
        }elseif ($sqltype=="update" || $sqltype=="insert" || $sqltype=="delete"){
            //判断如果 SQL 语句类型不为 SELECT，则执行如下操作
            if($rs)
                return true;                           //执行成功返回 true
            else
                return false;                          //否则返回 false
        }
     }
}
//分页类
class SepPage{
     var $rs;
     var $pagesize;
     var $nowpage;
     var $nowpages;
     var $array;
     var $conn;
     var $sqlstr;
     function ShowDate($sqlstr,$conn,$pagesize,$nowpage){  //定义方法
        if(!isset($nowpage) || $nowpage=="")           //判断变量值是否为空
            $this->nowpage=10;                         //定义每页输出的记录数
        else
```

```
            $this->nowpage=$nowpage;
        $this->pagesize=$pagesize;                    //定义每页输出的记录数
        $this->conn=$conn;                            //连接数据库返回的标识
        $this->sqlstr=$sqlstr;                        //执行的查询语句
        $this->rs=$this->conn->PageExecute($this->sqlstr,$this->pagesize,$this->nowpage);
        @$this->array=$this->rs->GetRows();           //获取记录数
            if(count($this->array)==0 || $this->rs==false)
                return false;
            else
                return $this->array;
    }
    function ShowPage($contentname,$utits,$anothersearchstr,$class,$page){
        $allrs=$this->conn->Execute($this->sqlstr);       //执行查询语句
        $record=count($allrs->GetRows());                 //统计记录总数
        $pagecount=ceil($record/$this->pagesize);         //计算共有几页
        $str.="共有".$contentname." ".$record." ".$utits." 每页显示 ".$this-> pagesize.
" ".$utits." 第 ".$this->rs->AbsolutePage()." 页/共 ".$pagecount." 页";
        $str.="    ";
        if(!$this->rs->AtFirstPage())
            $str.="<a  href=".$_SERVER['PHP_SELF']."?page=".$page."&pages=1".$anothersearchstr."  class=
".$class.">首页</a>";
        else
            $str.="<font color='#555555'>首页</font>";
        $str.=" ";
        if(!$this->rs->AtFirstPage())
            $str.="<a  href=".$_SERVER['PHP_SELF']."?page=".$page."&pages=".($this-> rs->AbsolutePage()-1).
$anothersearchstr." class=".$class.">上一页</a>";
        else
            $str.="<font color='#555555'>上一页</font>";
        $str.=" ";
        if(!$this->rs->AtLastPage())
            $str.="<a href=".$_SERVER['PHP_SELF']."?page=".$page."&pages=".($this->rs-> AbsolutePage()+1).
$anothersearchstr." class=".$class.">下一页</a>";
        else
            $str.="<font color='#555555'>下一页</font>";
        $str.=" ";
        if(!$this->rs->AtLastPage())
            $str.="<a href=".$_SERVER['PHP_SELF']."?page=".$page."&pages=".$pagecount. $anothersearchstr."
class=".$class.">尾页</a>";
        else
            $str.="<font color='#555555'>尾页</font>";
        if(count($this->array)==0 || $this->rs==false)
            return "";
        else
           return $str;
    }
}
?>
```

10.5.2　Smarty 模板配置类文件

在 Smarty 模板配置类文件中配置 Smarty 模板文件、临时文件、配置文件等文件路径。system.smarty.inc.php 文件的代码如下：

例程02　代码位置：光盘\TM\10\ShoppingSystem\system\system.smarty.inc.php

```
<?php
require("smarty/Smarty.class.php");                    //调用 Smarty 类文件
class SmartyProject extends   Smarty{                  //定义类，继承 Smarty 父类
      function SmartyProject(){                        //定义方法，配置 Smarty 模板
            $this->template_dir = "./";                //指定模板文件存储在根目录下
            $this->compile_dir = "./system/templates_c/"; //指定编译文件存储位置
            $this->config_dir = "./system/configs/";   //指定配置文件存储位置
            $this->cache_dir = "./system/cache/";      //指定缓存文件存储位置
      }
}
?>
```

10.5.3　类的实例化文件

在 system.inc.php 文件中，通过 require 语句包含 system.smarty.inc.php 和 system.class.inc.php 文件，执行类的实例化操作，并定义返回对象。完成数据库连接类的实例化后，调用其中的 GetConnId()方法连接数据库。system.inc.php 文件的代码如下：

例程03　代码位置：光盘\TM\10\ShoppingSystem\system\system.inc.php

```
<?php
require("system.smarty.inc.php");
require("system.class.inc.php");
$connobj=new ConnDB("mysql","localhost","root","root","db_shop",false); //数据库连接类实例化
$conn=$connobj->GetConnId();                            //连接数据库并返回连接标识
$admindb=new AdminDB();                                 //数据库操作类实例化
$seppage=new SepPage();                                 //分页类实例化
$smarty=new SmartyProject();                            //Smarty 模板配置类的实例化
?>
```

10.6　前台首页设计

10.6.1　前台首页概述

前台首页一般没有多少实质的技术，主要是加载一些功能模块，如登录模块、导航栏模块、公告栏模块等，使浏览者能够了解网站的内容和特点。首页的重要之处是要合理地对页面进行布局，既要尽可能地将重点模块显示出来，同时又不能因为页面凌乱无序，而让浏览者无所适从、产生反感。本

系统前台首页的运行结果如图 10.15 所示。



图 10.15　前台首页运行结果

10.6.2　前台首页技术分析

在前台首页中应用 switch 语句与 Smarty 模板中的 include 标签设计一个框架页面，实现不同功能模块在首页中的展示。

switch 语句在 PHP 动态文件中使用，根据超链接传递的值，包含不同的功能模块。

include 标签在 Smarty 模板页中使用，在当前模板页中包含其他模板文件。其语法如下：

```
{include file="file_name " assign=" " var="   "}
```

参数说明：file 指定包含模板文件的名称；assign 指定一个变量保存包含模板的输出；var 传递给待包含模板的本地参数，只在待包含模板中有效。

10.6.3　前台首页实现过程

前台首页使用的数据表：tb_commo、tb_public、tb_user、tb_links

（1）创建 index.php 动态页。在 index.php 动态页中，应用 include_once()语句包含相应的文件，应用 switch 语句，以超链接中参数 page 传递的值为条件进行判断，实现在不同页面之间跳转。index.php 的关键代码如下：

例程04　代码位置：光盘\TM\10\ShoppingSystem \index.php

```
<?php
session_start();                                   //初始化 session 变量
require_once("system/system.inc.php");             //获取数据库连接、管理和模板类返回的对象
include_once "public.php";                         //包含公告模块
include_once "links.php";                          //包含友情链接模块
```



```
if($_SESSION["member"]==""){                              //判断 session 变量的值是否为空
      $smarty->assign("member","F");                      //如果为空，则为模板变量赋值为 F
}else{
      $smarty->assign("member",$_SESSION['member']); //否则将 session 变量的值赋给模板变量
}
if($_GET["page"]==""){                                    //判断变量 page 的值是否为空
      $smarty->assign("page","F");                        //如果为空，为模板变量赋值为 F
}
switch($_GET["page"]){                                    //应用 switch 语句，根据条件进行判断，实现页面跳转
        case "hyzx":
           include_once "member.php";                     //包含指定的 PHP 文件
           if($_GET['action'] == 'modify'){               //根据 action 的值，指定不同的模板页
                $smarty->assign("switchs",'modifypwd.html');   //将指定的模板页赋给模板变量
           }else{
                $smarty->assign("switchs",'membershow.html');
           }
           break;
        case 'allpub':
          include_once 'allpub.php';
                $smarty->assign("switchs","allpub.html");  //将指定的模板页赋给模板变量
           break;
      …//省略了部分代码
        default:
          include_once 'nominate.php';
                $smarty->assign("switches","nominate.html");
           include_once 'newhot.php';
                $smarty->assign("switchs","newhot.html");
      }
      $smarty->display('index.html');                     //指定模板页
?>
```

（2）创建 index.html 模板页。在模板文件 index.html 中应用 Smarty 的 include 标签调用不同的模板文件，生成静态页面。关键代码如下：

例程05　代码位置：光盘\TM\10\ShoppingSystem\index.html

```
<table width="850" border="0" cellspacing="0" cellpadding="0">
  <tr>
    <td colspan="2">{include file="top.html"}</td>
  </tr>
  <tr>
    <td width="216" align="left" valign="top">
     {if $member=="F"}
          {include file="login.html"}
     {else}
          {include file="info.html"}
     {/if}
     {include file="public.html"}
     {include file="links.html"}
    </td>
    <td width="634" height="720" align="center" valign="top">
```

```
                {include file="search.html"}
                {include file=$switchs}
                {if $page==F}
                {include file=$switches}
                {/if}</td>
    </tr>
</table>
<table width="850" border="0" cellspacing="0" cellpadding="0">
        <tr>
                <td>{include file=buttom.html}</td>
        </tr>
</table>
```

说明

本系统的功能较多，结构比较复杂，对于初学者来说学起来可能会比较困难。所以，本书将系统中的各个功能模块所涉及的文件（如 PHP、TPL、CSS、JS 等）尽可能都单独实现。读者在学习其中某个模块时，可以将相关的文件统一放到同一个目录下单独测试。

10.7 登录模块设计

10.7.1 登录模块概述

用户登录模块是会员功能的窗口。匿名用户虽然也可以访问本网站，但只能进行浏览、查询等简单操作，而会员则可以购买商品，并且能享受超低价格。登录模块包括用户注册、用户登录和找回密码 3 部分，其运行结果如图 10.16 所示。

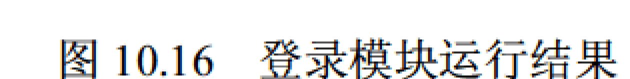


图 10.16　登录模块运行结果

10.7.2 登录模块技术分析

（1）Ajax 技术无刷新验证用户名是否被占用。关键代码如下：

例程06　代码位置：光盘\TM\10\ShoppingSystem\js\check.js

```
/*form 为传入的表单名称，本段代码为 register 表单*/
function chkname(form){
        /*如果 name 文本域的信息为空，则名为 name1 的 div 标签显示如下信息*/
        if(form.name.value==""){
                name1.innerHTML="<font color=#FF0000>请输入用户名！</font>";
        }else{
                /*否则获取文本域的值*/
                var user = form.name.value;
                /*生成 url 链接，将 user 的值传到 chkname.php 页进行判断*/
                var url = "chkname.php?user="+user;
                /*使用 xmlhttprequest 技术运行页面*/
                xmlhttp.open("GET",url,true);
                xmlhttp.onreadystatechange = function(){
```

```
		if(xmlhttp.readyState == 4){
			/*根据不同的返回值，在 div 标签中输出不同信息*/
			var msg = xmlhttp.responseText;
			if(msg == '3'){
				name1.innerHTML="<font color=#FF0000>用户名被占用！</font>";
				return false;
			}else if(msg == '2'){
				name1.innerHTML="<font color=green>恭喜您，可以注册!</font>";
			/*如果用户名正确，则将隐藏域的值改为 yes*/
				form.c_name.value = "yes";
			}else{
				name1.innerHTML="<font color=green>未知错误</font>";
			}
		}
	}
	xmlhttp.send(null);
	}
}
```

在该函数中调用 chkname.php 页，该页在会员登录时也会被调用，所以这里分两种情况，即有密码和无密码。无密码为注册验证，当没有返回结果时，说明该用户名可用；而有密码为登录验证，和无密码相反，只有查询记录存在时，才允许登录，并将用户名和用户 ID 存储到 session 中。chkname.php 页面的代码如下：

例程07　代码位置：光盘\TM\10\ShoppingSystem\ chkname.php

```
<?php
session_start()                                         //开启 session 支持
include_once 'system/system.inc.php';                   //加载数据库连接文件
$reback = '0';                                          //声明返回变量，初值为 0
$sql = "select * from tb_user where name='".$_GET['user']."'";  //生成 SQL 语句
/*下面的 SQL 语句加上了对密码的比较，这是用来验证登录的，因为登录验证使用的也是本页面*/
$password = $_GET['password'];
if(!empty($password)){
	$sql .= " and password = '".md5($password)."'";     //如果密码不为空，说明是登录验证
}
/*执行 SQL 语句*/
$rst = $conn->Execute($sql) or die('execute error');    //根据 name 字段查询数据库
if($rst->RecordCount() == 1){                           //判断 name 是否被使用
	$reback = '3';                                      //被使用返回 3
/*这里也是对登录页面进行的操作，当用户名和密码输入正确时，将用户名和 ID 存进 session 中*/
if(!empty($password)){
	$_SESSION['member'] = $rst->fields['name'];
	$_SESSION['id'] = $rst->fields['id'];
	$reback = '1';
	}
}else{
	$reback = '2';                                      //未被使用返回 2
}
echo $reback;
?>
```

（2）GD2 函数库生成验证码，关键代码如下：

例程08　代码位置：光盘\TM\10\ShoppingSystem\ yzm.php

```
<?php
srand((double)microtime()*1000000);
$im=imagecreate(42,16);
$black=imagecolorallocate($im,0,0,0);
$white=imagecolorallocate($im,255,255,255);
$gray=imagecolorallocate($im,200,200,200);
imagefill($im,0,0,$gray);
for($i=0;$i<4;$i++){
  $str=mt_rand(1,3);
  $size=mt_rand(3,6);
  $authnum=substr($_GET[num],$i,1);
  imagestring($im,$size,(2+$i*10),$str,$authnum,imagecolorallocate($im,rand(0,130),rand(0,130),rand(0,130)));
}
for($i=0;$i<200;$i++){
    $randcolor=imagecolorallocate($im,rand(0,255),rand(0,255),rand(0,255));
    imagesetpixel($im,rand()%70,rand()%30,$randcolor);
}
imagepng($im);
imagedestroy($im);
?>
```

10.7.3　用户注册

用户注册使用的数据表：tb_user

用户注册页面的主要功能是新用户注册。如果信息输入完整而且符合要求，则系统会将该用户信息保存到数据库中，否则显示错误原因，以便用户改正。用户注册页面的运行结果如图 10.17 所示。



图 10.17　用户注册

（1）创建 register.html 模板文件，编写用户注册页面。其中包含 createxmlhttp.js 和 check.js 两个 JS 脚本文件。其中，createxmlhttp.js 是 Ajax 的实例化文件，而 check.js 对用户注册信息进行验证，并且返回验证结果。

（2）创建 register.php 动态 PHP 文件，加载模板。register.php 文件的代码如下：

例程09　代码位置：光盘\TM\10\ShoppingSystem\register.php

```
<?php
        include_once 'system/system.inc.php';                //载入配置文件
        $smarty->assign('title','新用户注册');                  //设置 title
        $smarty->display('register.html');                    //显示 register.html
?>
```

（3）创建 reg_chk.php 文件，获取表单中提交的数据，将数据存储到指定的数据表中。reg_chk.php 文件的代码如下：



例程10　代码位置：光盘\TM\10\ShoppingSystem\reg_chk.php

```
<?php
	session_start();                                          //开启 session 支持
	include_once 'system/system.inc.php';                     //加载数据库连接文件
	/*获取表单信息*/
	$name = $_POST['name'];
	$password = md5($_POST['pwd1']);                          //使用 md5 给密码加密
	$question = $_POST['question'];
	$answer = $_POST['answer'];
	$realname = $_POST['realname'];
	$card = $_POST['card'];
	$tel = $_POST['tel'];
	$phone = $_POST['phone'];
	$Email = $_POST['email'];
	$QQ = $_POST['qq'];
	$code = $_POST['code'];
	$address = $_POST['address'];
	$addtime = $conn->DBDate(time());                         //使用 DBDate()函数生成当前时间
	/***************************/
	/*生成 SQL 语句*/
	$sql = " insert into tb_user( name, password, question, answer, realname, card, tel, phone, Email, QQ, code,
address, addtime ) ";
	$sql .= " values ('$name', '$password', '$question', '$answer', '$realname', '$card', '$tel', '$phone', '$Email', '$QQ',
'$code', '$address',$addtime)";
	/*执行 SQL 语句*/
	$rst = $conn->execute($sql);
	if($rst == false){
		echo '<script>alert(\'添加失败\');history.back;</script>';
	}else{
		$_SESSION['member'] = $name;
		$_SESSION['id'] = $conn->Insert_ID();
		echo "<script>top.opener.location.reload();alert('注册成功');window.close();</script>";
	}
?>
```

（4）创建“用户注册”超链接。当用户单击前台的用户注册按钮时，系统会调用 js 文件的 onClick 事件，弹出注册窗口。代码如下：

例程11　代码位置：光盘\TM\10\ShoppingSystem\login.html

```
<a href="#" id="login" onClick="reg()">用户注册</a>
```

这里使用到的 js 文件为 js/login.js，调用的函数为 reg()。该函数的代码如下：

例程12　代码位置：光盘\TM\10\ShoppingSystem\js\login.js

```
function reg(){
window.open("register.php", "_blank", "width=500,height=450",false);              //弹出新窗口
}
```

10.7.4 用户登录

用户注册使用的数据表：tb_user

用户登录模块的运行结果如图 10.18 所示，需要输入用户名、密码和验证码。

（1）创建模板文件 login.html，完成用户登录表单的设计。在该页面中当单击 Submit 按钮时，系统将调用 lg()函数对用户登录提交信息进行验证。lg()函数包含在 js/login.js 脚本文件内，代码如下：



图 10.18 用户登录页面

例程13 代码位置：光盘\TM\10\ShoppingSystem\js\login.js

```
function lg(form){
	/*检验用户名*/
	if(form.name.value==""){
		alert('请输入用户名');
		form.name.focus();
		return false;
	}
	/*检验密码*/
	if(form.password.value == "" || form.password.value.length < 6){
		alert('请输入正确密码');
		form.password.focus();
		return false;
	}
	/*检验验证码*/
	if(form.check.value == ""){
		alert('请输入验证码');
		form.check.focus();
		return false;
	}
	/*比较验证码*/
	if(form.check.value != form.check2.value){
		form.check.select();
		code(form);
		return false;
	}
	/*根据输入的用户名和密码生成 url*/
var user = form.name.value;
	var password = form.password.value;
	var url = "chkname.php?user="+user+"&password="+password;
	/*使用 xmlhttp 对象来返回验证结果*/
	xmlhttp.open("GET",url,true);
	xmlhttp.onreadystatechange = function(){
	if(xmlhttp.readyState == 4){
			var msg = xmlhttp.responseText;
			/*如果返回值不等于 2，说明用户名或密码错误*/
			if(msg != '2'){
				alert('用户名或密码错误!!');
				form.password.select();
```





```
                    form.check.value = '';
                    code(form);
                }else{
                /*如果成功，重新刷新本页*/
                    alert('欢迎光临');
                    location.reload();
                }
            }
    }
    xmlhttp.send(null);
    return false;
}
```

用户名和密码是在 chkname.php 页面中被验证。chkname.php 在 10.7.2 节中已经介绍，这里不再重复。

（2）创建用户信息模板文件 info.html。用户登录成功后，在原登录框位置将显示用户信息，用户可以通过"会员中心"对自己的信息做修改，也可以单击"查看购物车"超链接查看购物车商品，当用户离开时可以单击"安全离开"超链接。用户信息模块的主要代码如下：

例程14　代码位置：光盘\TM\10\ShoppingSystem\info.html

```
<!--显示当前登录用户名-->
欢迎您：{$member}
<!--会员中心超链接-->
<a href="?page=hyzx" id="info" class="lk">会员中心</a>
<!--查看购物车-->
<a href="?page=shopcar" class="lk">查看购物车</a>
<!--安全离开-->
<a onclick="javascript:logout()" style="cursor:hand" id="info">安全离开</a>
```

10.7.5　找回密码

用户注册使用的数据表：tb_user

登录模块的最后一个部分就是找回密码。找回密码是根据用户在填写资料时所填写的密保问题和密保答案来实现的。当用户单击"找回密码"超链接时，首先提示用户输入要找回密码的会员名称，然后根据密保问题填写密保答案，最后重新输入密码。找回密码模块的流程如图 10.19 所示。



第一步：输入会员名称



第二步：输入密保答案



第三步：更新密码



第四步：更改成功

图 10.19　找回密码流程图

1．创建模板文件

虽然找回密码需要 4 个步骤，但实际上每个步骤使用的都是相同的模板文件和 js 文件，只是被调用的表单和 js 函数略有差别。这里根据不同的文件来分别进行介绍。

该模板文件一共包含了 3 个表单，分别代表 3 个步骤，其核心代码如下：

例程15　代码位置：光盘\TM\10\ShoppingSystem\found.html

```
<!--载入两个 js 脚本文件-->
<script language="javascript" src="js/createxmlhttp.js"></script>
<script language="javascript" src="js/found.js"></script>
<!--第 1 个 div 标签-->
<div id="first">
<table width="200" border="0" cellspacing="0" cellpadding="0">
<form id="foundname" name="found" method="post" action="#">
  <tr><td> 找回密码</td></tr>
  <tr><td>会员名称：</td>
     <!--text 文本域，用于输入要找回密码的会员名称-->
<td><input id="user" name="user" type="text" class="txt"></td>
</tr>
  <tr><td>
<!--单击“下一步”按钮，能触发 onClick 事件来调用 chkname()函数-->
<input id = " next1 " name = " next1 " type = " button " class = " btn " value = " 下一步 " onClick = " return
chkname ( foundname ) "/></td></tr>
</form>
</table>
</div>
<!--第 2 个 div 标签，样式为隐藏-->
<div id="second" style="display:none;">
<table>
<form id="foundanswer" name="found" method="post" action="#">
  <tr><td > 找回密码</td></tr>
  <tr><td>密保问题：</td>
<!--用于显示密保问题的 div 标签-->
    <td <div id="question"></div></td></tr>
  <tr><td>密保答案：</td>
<!--文本域，用于填写密保答案-->
    <td ><input id="answer" name="answer" type="text" class="txt" /></td></tr>
  <tr>
<!--单击“下一步”按钮，用来触发 onClick 事件，并调用 chkanswer()函数-->
    <td><input id = " next2 " name = " next2 " type=" button " class=" btn " value =" 下一步 " onClick = " return
chkanswer ( foundanswer ) "></td>
    </tr>
</form>
</table>
</div>
<!--第 3 个 div 标签，样式也为隐藏，作用是修改密码-->
<div id='third' style="display:none;">
<table>
<form id="modifypwd" name="found" method="post" action="#">
  <tr><td> 输入密码</td></tr>
  <tr><td>输入密码：</td>
    <td><input id="pwd1" name="pwd1" type="password" class="txt"></td></tr>
  <tr><td>确认密码：</td>
    <td><input id="pwd2" name="pwd2" type="password" class="txt" /></td>
  </tr>
```

```
  <tr>
<!--单击“完成”按钮，调用 chkpwd()函数-->
        <td><input id = " mod " name = “ mod " type = " button " class = " btn " value = " 完成 " onClick = " return chkpwd (modifypwd) "></td>
      </tr>
</form>
</table>
</div>
```

可以看出，在上述 3 个表单中，只有一个表单默认情况下是显示的，其他则为隐藏。只有通过调用不同的 js 函数，才可以对其他表单进行操作。

2．创建 js 脚本文件

found.js 脚本文件包含 chkname()、chkanswer()和 chkpwd()3 个函数。其中，chkname()函数的作用是检查用户输入的会员名称，如果存在，则使用 xmlhttp 对象去调用生成的 url 进行处理判断。如果该用户存在，则隐藏当前表单，并显示下一个表单，最后输出密保问题。chkname()函数的代码如下：

例程16　代码位置：光盘\TM\10\ShoppingSystem\js\found.js

```
function chkname(form){
    /*获取文本框信息*/
    var user = form.user.value;
    /*如果为空，则输出提示*/
    if(user == ''){
        alert('请输入用户名');
        form.user.focus();
        return false;
    }else{
    /*否则，生成 url，并调用 xmlhttp 对象*/
        var url = "foundpwd.php?user="+user;
        xmlhttp.open("GET",url,true);
        xmlhttp.onreadystatechange = function(){
        if(xmlhttp.readyState == 4){
                var msg = xmlhttp.responseText;
                /*如果没有结果，则提示*/
                if(msg == '0'){
                    alert('没有该用户，请重新查找!');
                    form.user.select();
                    return false;
                }else{
                /*否则，隐藏当前表单，显示下个表单，并输出密保问题*/
                    document.getElementById('first').style.display = 'none';
                    document.getElementById('second').style.display = '';
                    document.getElementById('question').innerHTML = msg;
                }
            }
        }
        xmlhttp.send(null);
    }
}
```

其他两个函数也使用 xmlhttprequest 对象，实现方法相差无几，不同之处就是对返回值的处理，chkanswer()函数隐藏当前表单，显示下一个表单。chkanswer()函数的代码如下：

例程17　代码位置：光盘\TM\10\ShoppingSystem\js\found.js

```
function chkanswer(form) {
/*获取已隐藏表单数据*/
	var user = document.getElementById('user').value;
	/*获取密保答案*/
	var answer = form.answer.value;
	/*表单验证*/
	if(answer == ''){
		…
	}else{
		/*生成 url*/
		var url = "foundpwd.php?user="+user+"&answer="+answer;
		xmlhttp.open("GET",url,true);
		xmlhttp.onreadystatechange = function(){
			if(xmlhttp.readyState == 4){
				var msg = xmlhttp.responseText;
				if(msg == '0'){
					alert('问题回答错误');
					form.answer.select();
					return false;
				}else{
					document.getElementById('second').style.display = 'none';
					document.getElementById('third').style.display = '';
				}
			}
		}
		xmlhttp.send(null);
	}
}
```

而 chkpwd()函数则提示用户操作状态，如果成功，则关闭当前页。chkpwd()函数的代码如下：

例程18　代码位置：光盘\TM\10\ShoppingSystem\js\found.js

```
function chkpwd(form){
	/*获取名称和密码*/
	var user = document.getElementById('user').value;
	var pwd1 = form.pwd1.value;
	var pwd2 = form.pwd2.value;
	/*省略部分为表单验证*/
	var url = "foundpwd.php?user="+user+"&password="+pwd1;
	xmlhttp.open("GET",url,true);
	xmlhttp.onreadystatechange = function(){
		if(xmlhttp.readyState == 4){
			var msg = xmlhttp.responseText;
			if(msg == '1'){
				alert('密码修改成功，请重新登录');
				window.close();
```

```
                }
            }
        }
        xmlhttp.send(null);
}
```

3. 创建数据处理文件

foundpwd.php 文件的功能是根据用户输入的信息来检测数据表中的数据，并根据不同的输入信息返回不同的结果。该文件的代码如下：

例程19　代码位置：光盘\TM\10\ShoppingSystem\foundpwd.php

```
<?php
    include_once 'system/system.inc.php';                    //载入数据库连接文件
    $name= $_GET['user'];                                    //获取表单值
    $answer = $_GET['answer'];
    $password = $_GET['password'];
    $reback = '0';
    /*如果变量$answer 和$password 为空，说明用户只输入了会员名称*/
    if(empty($answer) && empty($password)){
        /*生成查询语句*/
        $namesql = "select * from tb_user where name = '".$name."'";
        /*获取查询结果*/
        $namerst = $conn->execute($namesql);
        /*如果查询记录为 1，说明输入正确*/
        if($namerst->recordCount() == 1){
            /*获取密保问题，并赋给变量$reback*/
            $question = $namerst->fields['question'];
            $reback = $question;
        }
    /*如果变量$answer 不为空，说明用户输入了密保答案*/
    }else if(!empty($answer)){
        /*根据密保答案和会员名称，生成查询语句*/
        $answersql = "select * from tb_user where name = '".$name."' and answer = '".$answer."'";
        /*返回查询结果，并判断是否有返回记录，如果有，说明输入正确*/
        $answerrst = $conn->execute($answersql);
        if($answerrst->recordCount() == 1){
            $reback = '1';
        }
    /*如果变量$password 不为空，说明用户输入了新密码*/
    }else if(!empty($password)){
        /*根据用户名称和新密码更新记录*/
        $sql = "select * from tb_user where name = '".$name."'";
        $arr = array();
        $rst = $conn->execute($sql);
        if($rst->RecordCount() == 1){
            $arr['password'] = md5($password);
            $updateSQL = $conn->GetUpdateSQL($rst,$arr,true);
            $conn->execute($updateSQL);
            $reback = '1';
```

```
            }
        }
        /*输出返回值*/
        echo $reback;
?>
```

4．加载模板页

因为所有登录模块的模板都不需要或者只需要传递一两个变量，所以 PHP 加载页的内容比较简单。找回密码页面的代码如下：

例程20　代码位置：光盘\TM\10\ShoppingSystem\found.php

```
<?php
        include_once 'system/system.inc.php';                  //载入配置文件
        $smarty->assign('title','找回密码');                     //向模板中传递 title 变量
        $smarty->display('found.html');                        //显示 found.html 模板页
?>
```

10.8　会员信息模块设计

10.8.1　会员信息模块概述

用户登录后，即可看到会员信息模块。在这里，可以进行查看或修改个人信息及密码、查看购物车和安全退出等操作。本节只对会员信息模块中的“会员中心”和“安全退出”进行讲解，关于“查看购物车”将在商品模块中进行介绍。会员信息模块的运行结果如图 10.20 所示。



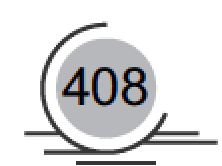
图 10.20　会员信息模块的运行结果

10.8.2　会员信息模块技术分析

在会员信息模块中，以 session 变量中存储的用户名称为条件，从会员信息表中查询出会员信息，并且将会员信息存储到模板变量中，最后在模板页中输出会员信息。member.php 文件的代码如下：

例程21　代码位置：光盘\TM\10\ShoppingSystem\member.php

```
<?php
        session_start();                                    //初始化 session 变量
        include_once 'system/system.inc.php';               //包含配置文件
        /*查找用户资料*/
        $sql = "select * from tb_user where name = '".$_SESSION['member']."'";          //执行查询操作
        $rst = $conn->execute($sql);
        $arr = $rst->GetArray();
```



```
    $smarty->assign('arr',$arr[0]);                    //将查询结果赋给模板变量
?>
```

member.php 文件中查询出的数据是会员信息模板功能实现的根本。

10.8.3　会员中心

用户注册使用的数据表：tb_user

当单击“会员中心”超链接时，会回传给当前页一个 page 值，当前页根据这个 page 值来载入 member.php 文件。

1．创建 PHP 页面

与登录模块设计不同，本节首先来创建 PHP 页面。因为该模块中的模板页需要使用数据库中的数据及一些动态信息，这些都需要在 PHP 页中先行获取及处理，然后再传给模板页。会员中心页面的代码请参考技术分析中的内容。

2．创建模板页

该模块包括查看信息模板及修改密码模板。首先介绍查看信息模板，代码如下：

例程22　代码位置：光盘\TM\10\ShoppingSystem\membershow.html

```
<!--显示超链接-->
<p>{$smarty.session.member}<a href = ' ?page=hyzx ' >查看信息</a><a href = ' ?page = hyzx&action=modify'
id="mem">修改密码</a></p>
<table>
<form id = " member " name = " member " method = " post " action = " modify_info_chk.php " onsubmit = " return
mem( member ) " >
<!--显示当前会员信息，此处为不可更改信息    -->
  <tr><td colspan="2">{$arr.name}信息（不可更改信息）</td></tr>
<!--因为格式相同，所以只显示一条，其他省略-->
  <tr><td>会员编号：</td><td > {$arr.id}</td></tr>
…
  <tr><td>{$arr.name}信息（可更改信息）</td></tr>
<!--可以修改的信息，放到表单元素中-->
  <tr><td>真实姓名：</td>
<td><input id="realname" name="realname" type="text" value="{$arr.realname}" /></td></tr>
  <tr><td>身份证号：</td>
    <td><input id="card" name="card" type="text" value="{$arr.card}" /></td></tr>
  <tr><td>移动电话：</td>
    <td><input id="tel" name="tel" type="text" value="{$arr.tel}"></td></tr>
  <tr><td>固定电话：</td>
    <td><input id="phone" name="phone" type="text" value="{$arr.phone}" /></td></tr>
  <tr><td>Email：</td>
    <td><input id="email" name="email" type="text" value="{$arr.Email}" /></td></tr>
  <tr><td>QQ 号：</td>
    <td><input id="qq" name="qq" type="text" value="{$arr.QQ}" /></td></tr>
  <tr>
    <td>邮编：</td>
    <td><input id="code" name="code" type="text" value="{$arr.code}" /></td></tr>
```

```
    <tr><td>地址：</td>
      <td><input id="address" name="address" type="text" value="{$arr.address}" /></td></tr>
    <tr>
<td><input name="enter" type="submit" id="enter" value="修改" />
            <input name="reset" type="reset" id="reset" value="重置" /></td></tr>
</form>
</table>
```

修改密码模板比信息模板要简单一些，因为不需要额外的传值。模板代码如下：

例程23　代码位置：光盘\TM\10\ShoppingSystem\modifypwd.html

```
<!--载入脚本文件-->
<script language="javascript" src="js/member.js"></script>
<p>{$smarty.session.member}<a href='?page=hyzx'>查看信息</a><a href='?page=hyzx&action=modify' >修改
密码</a></p>
<table>
<!--修改密码表单-->
<form id = " member " name = " member " method = " post " action = " modify_pwd_chk.php " onsubmit = " return
pwd (member) ">
    <tr><td >修改密码</td></tr>
    <tr><td >原密码：</td><td><input id="old" name="old" type="password" /></td></tr>
    <tr><td>新密码：</td><td><input id="new1" name="new1" type="password" /></td></tr>
    <tr><td>确认密码：</td><td><input id="new2" name="new2" type="password" /></td></tr>
    <tr><td><input id="enter" name="enter" type="submit" value="修改" /></td></tr>
</form>
</table>
```

3．创建脚本文件

该模块的脚本文件和用户注册模块类似，都是对信息的合法性进行验证，如信息是否为空、是否符合规范等，这里不再赘述。

4．创建处理页

当信息验证通过后，系统将跳转到处理页进行信息处理。本模块处理页分信息修改和密码修改两个页面。首先介绍信息修改页，代码如下：

例程24　代码位置：光盘\TM\10\ShoppingSystem\modify_info_chk.php

```
<?php
      session_start();                                                  //开启 session 支持
      include_once ‘system/system.inc.php’;                             //载入数据库连接文件
      $sql = 'select * from tb_user where id = '.$_SESSION['id'];       //根据 id 生成 SQL 查询语句
      $rst = $conn->execute($sql);                                      //执行 SQL 语句，返回查询结果集
      $mod = array();                                                   //声明数组
/*将修改信息添加到数组中*/
      $mod['realname'] = $_POST['realname'];
      $mod['card'] = $_POST['card'];
      $mod['tel'] = $_POST['tel'];
      $mod['phone'] = $_POST['phone'];
      $mod['Email'] = $_POST['email'];
```

```
    $mod['QQ'] = $_POST['qq'];
    $mod['code'] = $_POST['code'];
    $mod['address'] = $_POST['address'];
    /****************************************/
    $updateSQL = $conn->GetUpdateSQL($rst,$mod);                  //生成更新语句
    if($conn->execute($updateSQL))
        echo "<script>alert('修改成功');location=('index.php');</script>"; //修改成功，跳到首页
    else
        echo "<script>alert('修改失败');history.go(-1);</script>";     //否则，返回上一页
?>
```

密码修改页的操作流程也十分类似，只是更新的数组要小得多，只有一个字段。密码修改页的代码如下：

例程25　代码位置：光盘\TM\10\ShoppingSystem\modify_pwd_chk.php

```
<?php
    session_start();                                              //开启 session 支持
    include_once 'system/system.inc.php';;                        //载入数据库连接文件
    $oldpwd = md5($_POST['old']);                                 //对密码进行 md5 加密
    /*根据 id 和 md5 加密后的密码查询用户*/
    $sql = 'select * from tb_user where id = '.$_SESSION['id'].' and password = \''.$oldpwd.'\'';
    $rst = $conn->execute($sql);                                  //返回查询结果
    $mod = array();                                               //创建更新数组
    $mod['password'] = md5($_POST['new1']);                       //为新密码进行 md5 加密
    $updateSQL = $conn->GetUpdateSQL($rst,$mod);                  //生成更新语句
    if($conn->execute($updateSQL))
        echo "<script>alert('修改成功');location=('index.php');</script>"; //修改成功，返回首页
    else
        echo "<script>alert('修改失败');history.go(-1);</script>";     //否则，返回上一页
?>
```

10.8.4　安全退出

当用户需要离开网站时，可以单击“安全退出”超链接来调用 logout()函数，当用户确认退出后，则跳转到 logout 页面，销毁 session 并回到首页。安全退出所涉及的页面及代码如下：

例程26　代码位置：光盘\TM\10\ShoppingSystem\js\info.js

```
function logout(){
    if(confirm("确定要退出登录吗？ ")){                 //输出选择框，用户可以单击“确认”或“取消”按钮
        window.open('logout.php','_parent','',false);   //如果用户确认退出，则打开 logout.php 页
    }else
    return false;
}
```

例程27　代码位置：光盘\TM\10\ShoppingSystem\logout.php

```
<?php
    session_start();                                              //开启 session 支持
    session_destroy();                                            //销毁 session
```

```
    echo '<script>alert(\'用户已安全退出!\');location=(\'index.php\');</script>';        //回到首页
?>
```

10.9 商品显示模块

10.9.1 商品显示模块概述

本系统为用户提供了不同的商品展示方式，包括推荐商品、最新商品、热门商品等，能够使消费者有目的地选购商品。每个展示方式中包括商品的详细信息显示，为用户购买商品提供可靠的依据。本系统商品显示模块的运行结果如图 10.21 所示。

因为推荐商品、最新商品和热门商品的实现方法和过程基本相同，所以本节只讲解推荐商品模块。其他功能相关代码可参见光盘中的源程序。



图 10.21 商品显示模块页面

10.9.2 商品显示模块技术分析

商品显示功能实现的关键就是如何从数据库中读取商品信息，如何完成数据的分页显示。在定义 SQL 语句时，首先判断字段 isnom 的值，如果该字段为 1，即为推荐，否则为不推荐，然后再定义数据降幂排列，并设置每页显示 4 条记录，这就是完成商品显示的查询语句。代码如下：

例程28 代码位置：光盘\TM\10\ShoppingSystem\nominate.php

```
<?php
    include_once 'system/system.inc.php';                    //连接数据库
    $sql = "select id,name,pics,m_price,v_price from tb_commo where isnom = 1 order by id desc limit 4";
    $nomarr = $admindb->ExecSQL($sql,$conn);                 //执行查询语句
    $smarty->assign('nomarr',$nomarr);                       //将查询到的数据赋给模板变量
?>
```

最后，定义模板文件，通过 section 语句循环输出存储在模板变量中的数据，即完成商品展示的操作。section 是 Smarty 模板中的一个循环语句，该语句用于复杂数组的输出。其语法如下：

```
{section name="sec_name" loop=$arr_name start=num step=num}
```

参数含义：name 是该循环的名称；loop 为循环的数组；start 表示循环的初始位置，如 start=2，那么说明循环是从 loop 数组的第 2 个元素开始的；step 表示步长，如 step=2，那么循环一次后，数组的指针将向下移动两位，以此类推。



10.9.3　商品显示模块的实现过程

商品显示使用的数据表：tb_commo、tb_user

在 10.9.2 节中已经对商品显示所使用的技术、方法进行了概述，下面就介绍一下它具体的过程。

（1）创建 nominate.php 文件，从数据库中读取出推荐商品的数据，并将数据存储到模板变量中，其代码可以参考 10.9.2 节。

（2）创建 nominate.html 模板页，应用 section 语句输出商品信息，并添加相应的操作按钮或链接。模板页中一共有 3 个事件，即显示更多商品、查看商品和放入购物车。

☑ 当单击“更多商品”超链接时，将会重新加载本页面，并传递一个 page 变量。switch 语句会根据 page 值来显示。

☑ 当单击“查看详情”按钮时，将触发 onClick 事件，并将调用 openshowcommo()函数，同时，商品 id 会作为函数的唯一参数被传递进去。

☑ 当单击“购买”按钮时，同样会触发 onClick 事件，并调用 buycommo()函数，唯一的参数也是商品的 id。

商品模板页面的代码如下：

例程29　代码位置：光盘\TM\10\ShoppingSystem\nominate.html

```
<!--载入 css 样式-->
<link href="css/nominate.css" rel="stylesheet" type="text/css" />
<link href="css/links.css" rel="stylesheet" type="text/css" />
<!--载入 js 脚本文件-->
<script language="javascript" src="js/createxmlhttp.js"></script>
<script language="javascript" src="js/showcommo.js"></script>
<table width="643" border="0" cellpadding="0" cellspacing="0">
        <tr>
                <td colspan="6" background="images/default_14.gif" width="636" height="39" align="right" valign=
"middle"><a href="?page=nom" class="lk">&gt;&gt;more&lt;&lt;</a></td>
        </tr>
        <tr>
{section name=id loop=$nomarr}
                <td width="145" height="185" align="left" valign="top">
<table width="145" border="0" cellpadding="0" cellspacing="0" >
<tr>
        <td height="100" align="center" valign="middle"><img src="{$nomarr[id].pics}" width="100" height="80"
alt="{$nomarr[id].name}" style="border: 1px solid #f0f0f0;" ></td>
        </tr>
                <tr>
                <td height="32" align="center" valign="middle"><input id="showinfo" name="showinfo" type="button"
value="" class="showinfo" onclick="openshowcommo({$nomarr[id].id})"/> <input id="buy" name="buy"
type="button" value="" class="buy" onclick="return buycommo({$nomarr[id].id})" /></td>
        </tr>
        </table>
        </td>
        {/section}
        </tr>
</table>
```

（3）创建 showcommo.js 脚本文件。当单击“查看商品”按钮时，系统会弹出一个新的页面，并显示商品的详细信息；当单击“购买”按钮时，该商品将会被放到当前用户的购物车中，如果没有登录用户或商品已添加，则会提示错误信息。js 脚本文件的代码如下：

例程30　代码位置：光盘\TM\10\ShoppingSystem\js\showcommo.js

```
/*查看商品信息函数，将打开一个新页面*/
function openshowcommo(key){
	open('showcommo.php?id='+key,'_blank','width=560 height=300',false);
}
/*将购买商品添加到购物车中，将在 10.10 节中讲解*/
function buycommo(key){
	…
}
```

10.10　购物车模块设计

10.10.1　购物车模块概述

购物车在电子商务平台中是前台客户端程序中非常关键的一个功能模块。购物车的主要功能是保留用户选择的商品信息，用户可以在购物车内设置选购商品的数量，显示选购商品的总金额，还可以清除选择的全部商品信息，重新选择商品信息。购物车页面的运行结果如图 10.22 所示。

购物车模块主要实现添加商品、删除商品和更改数量等操作。

我的购物车						
	商品名称	购买数量	市场价格	会员价格	折扣率	合计
□	家庭影院	1	8888 元	6666 元	7.5	6666 元
□	数码相机	1	3200 元	2080 元	6.5	2080 元
全选 反选	删除选择		继续购物	去收银台		共计：8746 元

图 10.22　购物车页面

10.10.2　购物车模块技术分析

购物车功能实现最关键的部分就是如何将商品添加到购物车，如果不能完成商品的添加，那么购物车中的其他操作都没有任何意义。

在商品显示模块中，单击商品中的“购买”按钮，将商品放到购物车中，并进入到“购物车”页面。单击“购买”按钮调用 buycommo()函数，购买商品的 id 是该函数的唯一参数，在 buycommo()函数中通过 xmlhttp 对象调用 chklogin.php 文件，并根据回传值作出相应处理。buycommo()函数的代码如下：

例程31　代码位置：光盘\TM\10\ShoppingSystem\js\showcommo.js

```
/*
*添加商品，同时检查用户是否登录，商品是否重复等
*/
function buycommo(key){
	/*根据商品 id，生成 url*/
	var url = "chklogin.php?key="+key;
	/*使用 xmlhttp 对象调用 chklogin.php 页*/
	xmlhttp.open("GET",url,true);
```

```
	xmlhttp.onreadystatechange = function(){
		if(xmlhttp.readyState == 4){
				var msg = xmlhttp.responseText;
				/*用户没有登录*/
				if(msg == '2'){
					alert('请您先登录');
					return false;
				}else if(msg == '3'){
				/*商品已添加*/
					alert('该商品已添加');
					return false;
				}else{
				/*显示购物车*/
					location='index.php?page=shopcar';
				}
			}
	}
	xmlhttp.send(null);
```

在 chklogin.php 文件中将商品添加到购物车中。chklogin.php 页的代码如下：

例程32　代码位置：光盘\TM\10\ShoppingSystem\chklogin.php

```
<?php
	session_start();                                                //开启 session 支持
	/**
	  *1 表示添加成功
	  *2 表示用户没有登录
	  *3 表示商品已添加过
	  *4 表示添加时出现错误
	  *5 表示没有商品添加
	  */
	include_once 'system/system.inc.php';                           //载入数据库连接文件
	$reback = '0';                                                  //声明返回值
	if(empty($_SESSION['member'])){                                 //如果 session 值为空，说明没有登录
		$reback = '2';                                              //参见头注释
	}else{
		$key = $_GET['key'];                                        //获取商品 id
		if($key == ''){
			$reback = '5';
		}else{
			$id = (int)$_SESSION['id'];                             //获取登录会员 id 号
			$boo = false;                                           //设置一个布尔变量，初始为 false
			$addshop = array();                                     //数据库更新数组
			$sql = "select id,shopping from tb_user where id = ".$id;  //生成当前用户的查询语句
			$rst = $conn->execute($sql);                            //返回当前用户记录集
			$shopcont = $rst->fields['shopping'];                   //找到 shopping 字段
			if(!empty($shopcont)){                                  //如果字段不为空，说明已有商品
				$arr = explode('@',$shopcont);                      //将字段用“@”拆分并另存数组
				foreach($arr as $value){                            //循环输出数组内容
					if($key == $value[0]){                          //判断需要添加的商品是否存在
```

```
                        $reback = '3';                          //参见头注释
                        $boo = true;                            //同时布尔变量变为 true
                        break;                                  //停止循环
                    }
                }
                if(!$boo){                                      //如果布尔值没有改变，再添加商品
                    /*将商品数量默认设为 1，并连同商品 id 保存到字串中*/
                    $shopcont .= '@'.$key.',1';
                    /*将商品字串保存到更新数组中*/
                    $addshop['shopping'] = $shopcont;
                    /*生成更新语句*/
                    $updateSQL = $conn->GetUpdateSQL($rst,$addshop);
                    /*更新数据表*/
                    if(false == $conn->execute($updateSQL)){
                        $reback = '4';
                    }else{
                        $reback = '1';
                    }
                }
            }else{
                /*如果 shopping 字段为空，则直接更新*/
                $tmparr = $key.",1";
                $addshop['shopping'] = $tmparr;
                $updateSQL = $conn->GetUpdateSQL($rst,$addshop);
                if(false == $conn->execute($updateSQL)){
                    $reback = '4';
                }else{
                    $reback = '1';
                }
            }
        }
    }
    echo $reback;
?>
```

通过分析上述代码可知，shopping 字段保存的是购物车中的商品信息，一条商品信息包括两部分，即商品 id 和商品数量，其中商品数量默认为 1。两部分之间使用逗号“,”分隔，如果添加多个商品，则每个商品之间使用“@”分隔。

成功完成商品的添加操作后，即可进入到购物车页面，执行其他的操作。

10.10.3　购物车展示

购物车展示使用的数据表：tb_commo、tb_user

购物车页面分 PHP 代码页和 Smarty 模板页。在 PHP 代码页中，首先读取 tb_user 数据表中 shopping 字段的内容，如果字段为空，则输出“购物车中暂时没有商品！”；如果数据库中有数据，则循环输出数据，并将商品信息保存到数组中，再传给模板页。购物车页面的代码如下：

例程33　代码位置：光盘\TM\10\ShoppingSystem\myshopcar.php

```
<?php
	/*载入数据库连接文件和 Smarty 配置文件*/
	include_once 'system/system.inc.php';
	/*根据登录会员的用户名生成查询语句*/
	$sql1 = "select id,shopping from tb_user where name ='".$_SESSION['member']."'";
	/*执行查询语句返回结果集*/
	$rst = $conn->execute($sql1);
	/*如果字段为空，直接输出“购物车中暂时没有商品信息！”，并结束运行*/
	if($rst->fields['shopping'] == ''){
		echo "<p>";
		echo '购物车中暂时没有商品!';
		exit();
	}
	/*如果有结果集，则将结果集另存为数组*/
	$tmparr = $rst->GetAssoc();
	$commarr = array();
	/*输出 shopping 字段值*/
	foreach($tmparr as   $value){
		/*使用“@”进行分隔，得到数组*/
		$tmpnum = explode('@',$value);
		/*得到购买商品种类*/
		$shopnum = count($tmpnum);
		/*商品总价值*/
		$sum = 0;
		/*数组循环*/
		foreach($tmpnum as $key => $vl){
			/*使用逗号“,”拆分单一的商品信息*/
			$s_commo = explode(',',$vl);
			/*根据保存的商品信息查找商品*/
			$sql2 = "select id,name,m_price,fold,v_price from tb_commo";
			$commsql = $sql2." where id = ".$s_commo[0];
			/*返回结果集*/
			$commrst = $conn->execute($commsql);
			/*将结果集另存数组*/
			$arr = $commrst->GetArray();
			/*将需要显示的信息处理后保存到数组中*/
			$arr[0]['num'] = $s_commo[1];
			$arr[0]['total'] = $s_commo[1]*$arr[0]['v_price'];
			$sum += $arr[0]['total'];
			$commarr[$key] = $arr[0];
		}
	}
	/*将得到的两个数组和总消费金额传递给模板*/
	$smarty->assign('shoparr',$shopnum);
	$smarty->assign('commarr',$commarr);
	$smarty->assign('sum',$sum);
	$smarty->assign('title','我的购物车');
?>
```

商品的模板页不仅要负责用户购买商品信息的输出，而且还要提供可以对商品进行修改、删除等操作的事件接口。模板页代码如下：

例程34 代码位置：光盘\TM\10\ShoppingSystem\myshopcar.html

```
<!--载入使用到的 js 脚本文件-->
<script language="javascript" src="js/createxmlhttp.js"></script>
<script language="javascript" src="js/shopcar.js"></script>
<table>
<!--购物车表单-->
<form id="myshopcar" name="myshopcar" method="post" action="#">
  <tr><td>我的购物车</td></tr>
  <tr><td > </td><td>商品名称</td><td>购买数量</td><td>市场价格</td>
<td>会员价格</td><td>折扣率</td><td>合计</td></tr>
<!--使用 foreach 表单循环输出商品种类-->
{foreach key=key item=item from=$commarr}
<!--删除复选框，名称用数组表示，每个复选框的值为商品的 id 号-->
  <tr><td><input id="chk" name="chk[]" type="checkbox" value="{$item.id}"></td>
<!--显示该商品的名称-->
     <td><div id = "c_name{$key}">  {$item.name}</div></td>
<td>
<!--商品数量文本框，每个复选框的名称为"cnum"+key 值-->
     <input id = " cnum{$key} " name = " cnum{$key} " type = " text " value = " {$item.num} " onkeyup = " cvp
( {$key},{$item.v_price},{$shoparr})"></td>
<!--商品的其他信息-->
    <td><div id="m_price{$key}"> {$item.m_price}</div></td>
    <td><div id="v_price{$key}"> {$item.v_price}</div></td>
    <td><div id="fold{$key}"> {$item.fold}</div></td>
    <td><div id="total{$key}"> {$item.total}</div></td>
  </tr>
<!--循环结束-->
{/foreach}
  <tr>
<td colspan="3">
<!--全选、反选和删除操作-->
      <a href="#" onclick="return alldel(myshopcar)">全选</a>
      <a href="#" onclick="return overdel(myshopcar);">反选</a>
      <input type="button" value="删除选择" onClick = 'return del(myshopcar);'></td>
<td>
<!--继续购物-->
     <input type="button" value="继续购物" onclick="return conshop(myshopcar)" /></td>
<td>
<!--将当前用户名保存到隐藏域中-->
     <input id="uid" name="uid" type="hidden" value="{$smarty.session.member}" >
<!--结算，去收银台-->
     <input type="button" class="btn" value="去收银台" onclick="return formset(form)" /></td>
    <td colspan="2"><div id='sum'>共计：{$sum} 元</div></td>
    </tr>
</form>
</table>
```

10.10.4　更改商品数量

更改商品数量使用的数据表：tb_commo、tb_user

对于新添加的商品，默认的购买数量为 1，在购物车页面可以对商品的数量进行修改。当商品数量发生变化时商品的合计金额和商品总金额会自动发生改变，该功能是通过触发 text 文本域的 onkeyup 事件调用 cvp()函数实现。cvp()函数有 3 个参数，分别是商品 id、商品单价和商品类别。

首先，通过商品的 id 可以得到要修改商品的相关表单和标签属性。然后，通过商品单价和输入的商品数量计算该商品的合计金额。接着，使用 for 循环得到其他商品的合计金额。最后，将所有的合计金额累加，并输出到购物车页面。cvp()函数的代码如下：

例程35　代码位置：光盘\TM\10\ShoppingSystem\js\shopcar.js

```
/*自动刷新总金额
 * key：商品 id
 * vpr：商品单价
 * shoparr：商品种类数
*/
function cvp(key,vpr,shoparr){
	var n_pre = 'total';                                  //div 标签的 id 前缀
	var num = 'cnum'+key.toString();                      //根据 key 值生成文本域的 id 值
	var total = n_pre+key.toString();                     //根据 key 值生成 div 标签的 id 值
	var t_number = document.getElementById(num).value;    //获取输入的商品数量
	var ttl = t_number * vpr;                             //根据商品数量和单价，计算商品金额
	document.getElementById(total).innerHTML = ttl;       //更新商品总金额
	var sm = 0;                                           //所有商品的总金额，初始为 0
	for(var i = 0; i < shoparr; i++){                     //根据商品种类数，循环得到其他商品金额
		var aaa = document.getElementById(n_pre+i.toString()).innerText;
		sm += parseInt(aaa);                              //将所有商品金额累加
	}
	document.getElementById('sum').innerHTML = '共计：'+sm+'  元';
}
```

这里所更改的商品数量，并没有被保存到数据库中，如果希望保存，那么单击“继续购物”按钮，则可以将商品数量更新到数据库中。

10.10.5　删除商品

删除商品使用的数据表：tb_commo、tb_user

当对添加的商品不满意时，可以对商品进行删除操作。操作流程为：首先选中要删除的商品前面的复选框，如果全部删除，则可以单击“全选”或“反选”按钮；然后单击“删除选择”按钮，在弹出的警告框中单击“确定”按钮，商品将被全部删除。删除商品的页面结果如图 10.23 所示。

所有的删除操作都是通过 js 脚本文件 shopcar.js 来实现，相关的函数包括 alldel()函数、overdel()函数和 del()函数。

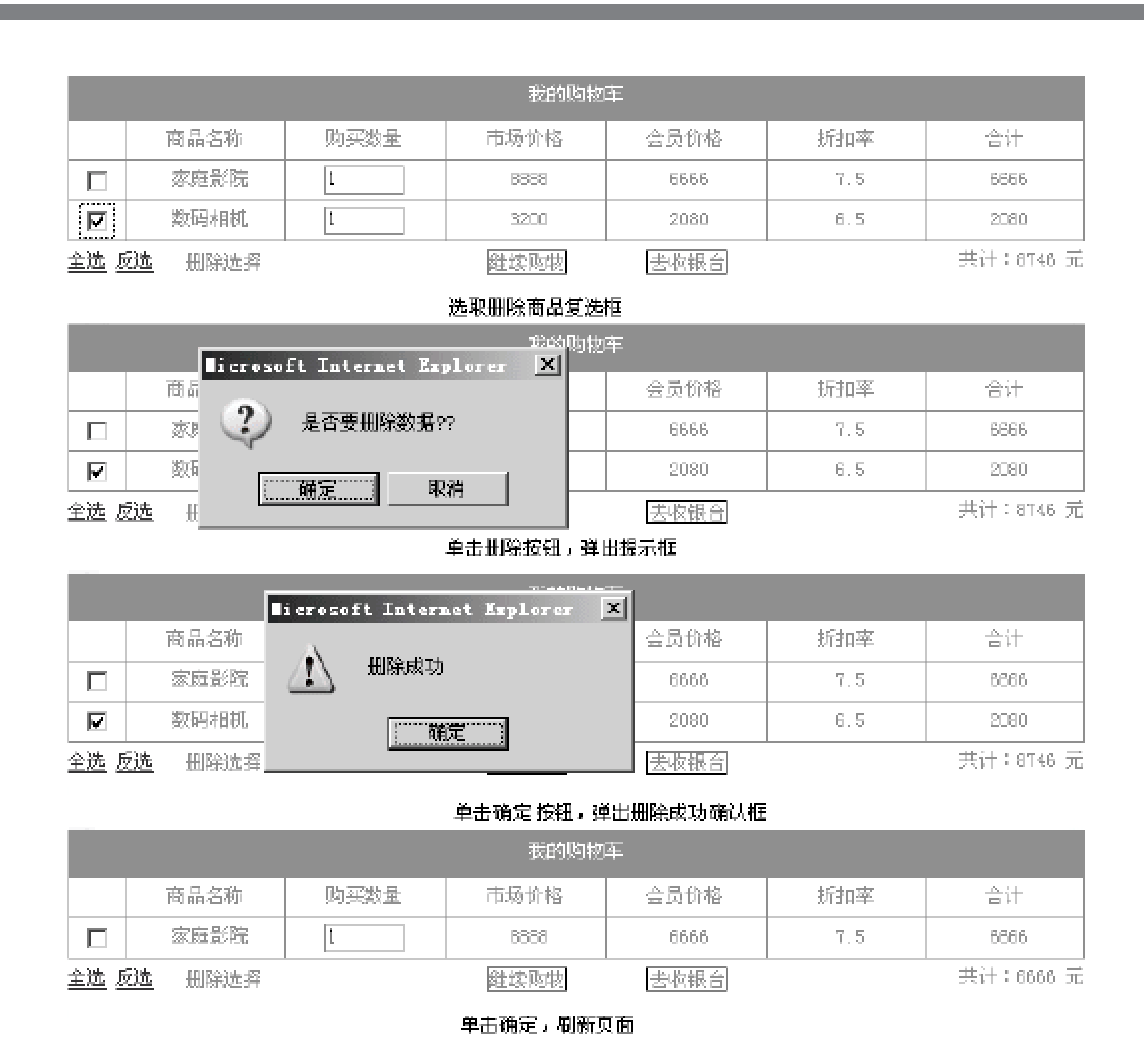


图 10.23　删除商品的流程

alldel()函数和 overdel()函数实现的原理比较简单，通过触发 onClick 事件来改变复选框的选中状态。代码如下：

例程36　代码位置：光盘\TM\10\ShoppingSystem\js\shopcar.js

```
/*全部选择*/
function alldel(form){
	var leng = form.chk.length;                          //获取复选框数量
	if(leng==undefined){                                 //如果等于 undefined，说明只有一个复选框
	   if(!form.chk.checked)
	          form.chk.checked=true;                     //将复选框置于选中状态
	 }else{                                              //否则，说明有多个复选框
	   for( var i = 0; i < leng; i++)                    //使用 for 循环，将所有复选框选中
	     {
	          if(!form.chk[i].checked)
	               form.chk[i].checked = true;
	     }
	 }
	return false;
}
/*反选，就是将选中的复选框取消选中，而未被选中的复选框则被选中*/
function overdel(form){
	 var leng = form.chk.length;                         //获取复选框个数
	 if(leng==undefined){                                //如果 leng 为 undefiend，说明只有一个复选框
	   if(!form.chk.checked)
	          form.chk.checked=true;
	     else
	          form.chk.checked=false;
	 }else{                                              //否则，说明有多个复选框
	   for( var i = 0; i < leng; i++)                    //使用 for 循环，将所有复选框操作
	     {
	          if(!form.chk[i].checked)                   //根据复选框 checked 属性的情况，进行反向选择
	               form.chk[i].checked = true;
```

```
				else
					form.chk[i].checked = false;
			}
		}
	return false;
}
```

使用 alldel()或 overdel()选中复选框后，即可调用 del()函数来实现删除功能。del()函数首先使用 for 循环，将被选中复选框的 value 值取出并存成数组，然后根据数组生成 url，并使用 xmlhttp 对象调用这个 url，当处理完毕后，根据返回值弹出提示或刷新本页。该函数的代码如下：

例程37　代码位置：光盘\TM\10\ShoppingSystem\js\shopcar.js

```
/*删除记录*/
function del(form){
	if(!window.confirm('是否要删除数据??')){

	}else{
		var leng = form.chk.length;                         //返回复选框的长度
		if(leng==undefined){
			if(!form.chk.checked){
					alert('请选取要删除数据!');
			}else{                                          //如果只有一个复选框，且处于选中状态
				rd = form.chk.value;                        //将复选框的 value 值直接赋给变量 rd
				var url = 'delshop.php?rd='+rd;             //根据 rd 生成 url
				xmlhttp.open("GET",url,true);               //调用 xmlhttp 对象
				xmlhttp.onreadystatechange = delnow;        //调用 delnow()函数
				xmlhttp.send(null);
			}
		}else{                                              //如果复选框为多个
			var rd=new Array();                             //声明数组
			var j = 0;
			for( var i = 0; i < leng; i++)                  //循环检查复选框状态
			{
				if(form.chk[i].checked){
					rd[j++] = form.chk[i].value;            //将被选中复选框的 value 值存到 rd 内
				}
			}
			if(rd == ''){
				alert('请选取要删除数据!');
			}else{
				var url = "delshop.php?rd="+rd;             //根据 rd 生成 url
				xmlhttp.open("GET",url,true);
				xmlhttp.onreadystatechange = delnow;
				xmlhttp.send(null);
			}
		}
	}
	return false;
}
/*显示状态*/
```

```
function delnow(){
	if(xmlhttp.readyState == 4){
		if(xmlhttp.status == 200){
			var msg = xmlhttp.responseText;			//获取 xmlhttp 对象返回的文本值
			if(msg != '1'){
				alert('删除失败');				//如果为 1，说明删除失败
			}else{
				alert('删除成功');
				location=('?page=shopcar');		//否则，说明删除成功，刷新购物车
			}
		}
	}
}
```

10.10.6 保存购物车

保存购物车使用的数据表：tb_commo、tb_user

当用户希望保存商品更改后的商品数量时，可以单击“继续购物”按钮，将触发 onClick 事件调用 conshop()函数保存数据，该函数有一个参数，就是当前表单的名称。在 conshop()函数内，根据复选框和商品数量文本域，生成两个数组 fst 和 snd，分别保存商品 id 和商品数量。

这里要注意，两个数组的值是要相互对应的，如商品 1 的 id 保存到 fst[1]中，那么商品 1 的数量就要保存到 snd[1]中，然后根据这两个数组生成一个 url，使用 xmlhttprequest 对象调用 url，最后根据回传信息作出相应的判断。conshop()函数的代码如下：

例程38 代码位置：光盘\TM\10\ShoppingSystem\js\shopcar.js

```
/*更改商品数量*/
function conshop(form){
	var n_pre = 'cnum';						//商品数量文本域前缀
	var lang = form.chk.length;				//获取复选框的数量
/*如果只有一个复选框，那么将商品 id 和商品数量直接保存到变量中*/
	if(lang == undefined){
		var fst = form.chk.value;
		var snd = form.cnum0.value;
/*否则，将商品 id 和对应的商品数量保存到两个相应的数组中*/
	}else{
		var fst= new Array();				//商品 id 数组
		var snd = new Array();				//商品数量数组
/*循环获取复选框的 value 值和商品数量文本框的 value 值*/
		for(var i = 0; i < lang; i++){
			var nm = n_pre+i.toString();
			var stmp = document.getElementById(nm).value;
/*对商品数量文本框判断，不允许为空，而且不允许为非数字输入*/
			if(stmp	== '' || isNaN(stmp)){
				alert('不允许为空，必须为数字');
				document.getElementById(nm).select();
				return false;
			}
			snd[i] = stmp;
```



```
                var ftmp = form.chk[i].value;
                fst[i] = ftmp;
            }
        }
        var url = 'changecar.php?fst='+fst+'&snd='+snd;            //生成 url
        xmlhttp.open("GET",url,true);                              //调用 xmlhttp 对象
        xmlhttp.onreadystatechange = updatecar;                    //调用 updatecar()函数
        xmlhttp.send(null);
}
/*对 xmlhttp 对象的返回值进行处理*/
function updatecar(){
        if(xmlhttp.readyState == 4){
            var msg = xmlhttp.responseText;
                if(msg == '1'){
                    location='index.php';                          //如果操作成功，返回首页
                }else{
                    alert('操作失败');                               //否则，提示错误
                }
        }
}
```

在 conshop()函数中调用的 changecar.php 页为数据处理页，该页将商品 id 和商品数量进行重新排列，并保存到 shopping 字段内。该页面的代码如下：

例程39　代码位置：光盘\TM\10\ShoppingSystem\changecar.php

```
<?php
        include_once 'system/system.inc.php';                                    //连接数据库
        $sql = 'select id,shopping from tb_user where id = '.(int)$_SESSION['id'];  //生成 SQL 查询语句
        $fst = $_GET['fst'];                                                      //获取商品 id
        $snd = $_GET['snd'];                                                      //获取商品数量
        $reback = '0';                                                            //设置返回值，默认为 0
        $rst = $conn->execute($sql);                                              //执行查询语句，返回查询结果集
        $changecar = array();                                                     //创建更新数组
        if($fst != '' and $snd != ''){
            /*使用逗号“,”将商品 id 和商品数量分隔，并分别保存到两个临时数组中*/
            $farr = explode(',',$fst);
            $sarr = explode(',',$snd);
            /*临时数组*/
            $upcar = array();
            /*使用 for 循环，将商品 id 和商品数量一一对应，重新组合*/
            for($i = 0; $i < count($farr); $i++){
                $upcar[$i] = $farr[$i].','.$sarr[$i];
            }
            /*如果数组大于 1，说明有多个商品，使用“@”将不同商品分隔开*/
            if(count($farr) > 1){
                $changecar['shopping'] = implode('@',$upcar);
            }else{
            /*如果只有一个商品，则直接保存到更新数组中*/
                 $changecar['shopping'] = $upcar[0];
            }
            /*生成更新语句*/
```

```
                $UpdateSql = $conn->GetUpdateSQL($rst,$changecar,true);
           /*更新数据库*/
                if(false == $conn->execute($UpdateSql)){
                     $reback = '2';
                }else{
                     $reback = '1';
                }
     }
     echo $reback;
?>
```

10.11　收银台模块设计

10.11.1　收银台模块概述

当用户停止浏览商品准备结账时，可以单击购物车页面中的“去收银台”按钮，该按钮将触发 onClick 事件调用 formset()函数显示订单页面，当用户提交订单后，系统将订单保存到数据表 tb_form 中，同时清空购物车，并显示订单信息提醒用户记录订单号。当货款发出后，还可以对订单进行查询。收银台页面的运行结果如图 10.24 所示。

本节所涉及的页面有显示订单（formset()函数）、填写订单（settle.php、settle.html）、处理订单（settle_chk.php）、反馈订单（forminfo.php、forminfo.html）和查询订单 5 部分。



图 10.24　收银台页面的运行结果

10.11.2　收银台模块技术分析

在收银台模块中，通过 ADODB 中的方法完成订单信息的添加和数据的更新操作。在 ADODB 中，数据的添加和更新可以采用两种方法：第一种直接定义 INSERT 或者 UPDATE 语句，通过 execute()函数来执行添加或者更新的操作；第二种使用 ADODB 提供的函数来完成。

1．GetInsertSQL()函数

GetInsertSQL()函数向数据库中添加新的记录，其语法如下：

```
GetInsertSQL(&$rs, $arrFields)
```

参数$rs 设置要添加记录的结果集；参数$arrFields 设置新记录的字段值及字段名称。

2．GetUpdateSQL()函数

GetUpdateSQL()函数更新数据库中指定的记录，其语法如下：

```
GetUpdateSQL(&$rs, $arrFields, $forceUpdate=false)
```

参数$rs 指定要更新的结果集；参数$arrFields 指定更新的字段和内容；参数$forceUpdate 控制更新





操作是否执行，如果$forceUpdate 为 true，无论$arrFields 和$rs 的值是否相等都将更新。

10.11.3　显示订单

显示订单使用的数据表：tb_commo、tb_user

订单信息提交页面的输出由 formset()函数决定，它将商品信息进行整理，然后通过 open()方法打开 settle.php 页来显示订单，并将整理后的商品信息传递到 settle.php 文件中。formset()函数的代码如下：

例程40　代码位置：光盘\TM\10\ShoppingSystem\js\shopcar.js

```
function formset(form){
var uid = form.uid.value;                          //获取订单提交人的名称
var n_pre = 'cnum';                                //商品数量文本前缀
    var lang = form.chk.length;                    //复选框个数
    /*如果复选框只有一个*/
    if(lang == undefined){
        var fst = form.chk.value;                  //商品 id
        var snd = form.cnum0.value;                //商品数量
    /*当有多个复选框时*/
    }else{
        var fst= new Array();
        var snd = new Array();
    /*使用 for 循环，获取商品数量和商品 id*/
        for(var i = 0; i < lang; i++){
            var nm = n_pre+i.toString();
            var stmp = document.getElementById(nm).value;
            if(stmp   == '' || isNaN(stmp)){
                alert('不允许为空，必须为数字');
                document.getElementById(nm).select();
                return false;
            }
            snd[i] = stmp;
            var ftmp = form.chk[i].value;
            fst[i] = ftmp;
        }

    }
/*使用 open()方法打开 settle.php 页，并将当前用户会员名称、商品 id 和商品数量的值传过去*/
open('settle.php?uid='+uid+'&fst='+fst+'&snd='+snd,'_blank','width=420 height=220',false);
}
```

说明

因为 open()方法使用了_blank 参数来打开一个新的页面，session 值传不过去，所以这里使用隐藏域来传递用户名称。

10.11.4 填写订单

填写订单使用的数据表：tb_commo、tb_user、tb_form

settle.php 直接将接收的值传给 settle.html 模板，并载入 settle.html 模板。settle.php 页面的代码如下：

例程41 代码位置：光盘\TM\10\ShoppingSystem\settle.php

```
<?php
    include_once 'system/system.inc.php';                //载入 Smarty 配置文件
    $fst = $_GET['fst'];                                  //获取 GET 属性值
    $snd = $_GET['snd'];
    $uid = $_GET['uid'];
    $smarty->assign('title','结账');                      //将标题、数组等数据传递给模板
    $smarty->assign('fst',$fst);
    $smarty->assign('snd',$snd);
    $smarty->assign('uid',$uid);
    $smarty->display('settle.html');                      //显示 settle.html 模板文件
?>
```

settle.html 模板显示一个表单，这个表单的内容需要用户来填写，包括收货人、联系电话等信息。而从 PHP 页传过来的几个变量则被保存到隐藏域以传递到处理页，在表单中将数据提交到 settle_chk.php 处理页。

10.11.5 处理订单

处理订单使用的数据表：tb_commo、tb_user、tb_form

处理页 settle_chk.php 获取表单中提交的数据，根据用户提交的商品信息，重新查找数据表 tb_commo，并从数据表中提取商品信息，保存到数组中，然后处理页将数组作为一条记录添加到表 tb_form 内。

数据添加成功的同时，处理页会根据 uid 找到该用户，将 shopping 字段清空，最后调用 forminfo.php 页来显示新添加的订单信息。settle_chk.php 页的代码如下：

例程42 代码位置：光盘\TM\10\ShoppingSystem\settle_chk.php

```
<?php
/*获取表单值，并存储到添加数组中*/
    include_once 'system/system.inc.php';
    $sql = "select * from tb_form where id = -1";
    $rst = $conn->execute($sql);
    $addform = array();
    $addform['vendee'] = $_POST['uid'];                      //提交订单用户
    $addform['commo_id'] = $_POST['fst'];                    //购买商品 id
    $addform['commo_num'] = $_POST['snd'];                   //购买商品数量
/*根据商品 id 和商品数量，获取所需数据*/
    $addform['formid'] = time()                              //订单号
    $tmpid = explode(',',$addform['commo_id']);              //将商品 id 存为数组
    $tmpnm = explode(',',$addform['commo_num']);             //将商品数量存为数组
    $number = count($tmpid);                                 //获取商品数组长度
    /*当购买商品种类大于 1 时*/
    if($number >1){
```

```
		$tmpna = array();								//商品名称数组
		$tmpvp = array();								//商品单价数组
		$tmpfd = array();								//商品折扣数组
		$tmptt = 0;									//商品总额数组
		/*使用 for 循环获取各种商品的信息*/
		for($i = 0; $i < $number; $i++){
			/*生成查询语句，执行 SQL 语句，最后返回查询结果集*/
			$tmpsql = "select name,v_price,fold from tb_commo where id = ".$tmpid[$i];
			$tmprst = $conn->execute($tmpsql);
			/*将每种商品的信息放到对应的数组中*/
			$tmpna[$i] = $tmprst->fields['name'];
			$tmpvp[$i] = $tmprst->fields['v_price'];
			$tmpfd[$i] = $tmprst->fields['fold'];
			$tmptt += $tmprst->fields['v_price'] * $tmpnm[$i];
		}
		/*将获取的信息数组分别生成长字符串，数组值之间使用逗号“,”分隔*/
		$addform['commo_name'] = implode(',',$tmpna);				//商品名称
		$addform['agoprice'] = implode(',',$tmpvp);				//商品价格
		$addform['fold'] = implode(',',$tmpfd);					//商品折扣
		$addform['total'] = $tmptt;							//商品总额
	/*当购买一种商品时*/
	}else if($number == 1){
		$tmpsql = "select name,v_price,fold from tb_commo where id = ".$tmpid[0];
		$tmprst = $conn->execute($tmpsql);
		$addform['commo_name'] = $tmprst->fields['name'];
		$addform['agoprice'] = $tmprst->fields['v_price'];
		$addform['fold'] = $tmprst->fields['fold'];
		$addform['total'] = $tmprst->fields['v_price'] * $tmpnm[0];
	/*没有商品时*/
	}else{
		echo 'error';
		exit();
	}
	/*将表单的其他信息存入添加数组*/
	$addform['taker'] = $_POST['taker'];						//收货人
	$addform['code'] = $_POST['code'];						//邮编
	$addform['tel'] = $_POST['tel'];						//联系电话
	$addform['address'] = $_POST['address'];					//地址
	$addform['del_method'] = $_POST['del'];					//送货方式
	$addform['pay_method'] = $_POST['pay'];					//付款方式
	$addform['state']   = '0';							//订单状态，0 为未处理
	$InsertSQL = $conn->GetInsertSQL($rst,$addform);			//生成 SQL 添加语句
	if(false == $conn->execute($InsertSQL)){
		/*如果购买失败，返回上层*/
		echo "<script>alert('购买失败');history.back;</script>";
	}else{
		/*如果成功，删除购物车信息，并显示订单信息*/
		$updsql = "select * from tb_user where name = '".$_POST['uid']."'";
		$updrst = $conn->execute($updsql);
		$arr = array();
		$arr['consume'] = $addform['total'];
		$arr['shopping'] = '';
```

```
            $UpdateSQL = $conn->GetUpdateSQL($updrst,$arr);
            $conn->execute($UpdateSQL);
            echo "<script>top.opener.location.reload();</script>";
            echo "<script>open('forminfo.php?fid=$fid','_blank','width=600 height=450',false);</script>";
            echo "<script>window.close(); </script>";
        }
?>
```

由于篇幅所限，有关反馈订单和查询订单的内容这里不再讲解，请读者参考本书光盘中的源代码。

10.12　后台首页设计

10.12.1　后台首页概述

后台管理系统是网站管理员对商品、会员及公告等信息进行统一管理的场所，本系统的后台主要包括以下功能。

☑　类别管理模块：主要包括对商品类别的添加、修改及删除操作。

☑　商品管理模块：主要包括对商品的添加、修改、删除及订单处理。

☑　用户管理模块：主要包括管理员管理和会员管理。其中，管理员管理是实现对管理员的添加、删除和修改功能；会员管理则包括删除和冻结功能。

☑　公告管理模块：主要包括公告的添加及删除操作。

☑　链接管理模块：主要包括添加、修改和删除友情链接。

后台首页的运行结果如图 10.25 所示。



图 10.25　后台首页的运行结果

10.12.2　后台首页技术分析

对后台首页的设计和前台首页不同，其使用的是框架布局。框架布局的特点是：可以将容器窗口划分为若干个子窗口，每个子窗口可以分别显示不同的网页，网页之间为相互独立的，没有直接的关联，又由一个网页将这些分开的网页组成一个完整的网页，显示在浏览者的浏览器中。框架布局的好处是：每次浏览者发出对页面的请求时，只下载发生变化的框架页面，其他子页面保持不变。下面来具体看一下框架布局的使用格式及属性。

1．框架布局格式

框架布局的格式很简单，只要几行代码即可，常用的格式如下：

```
<html>
<head>
…
</head>
<frameset>
     <frame>
     <frame>
</frameset>
<noframes>
      <body>
      …
      </body>
</noframes>
</html>
```

其中<frameset>和<frame>标签是框架集标记，而<noframes>标签是为了防止浏览器不支持框架而实行的一种补救措施。如果浏览器不支持框架集，就会执行<noframes>标记中的内容，让用户能够正常浏览网页。

2．框架集属性

框架集包含各个框架的信息，通过<frameset>标记来定义。框架是按照行和列来组织的，可以使用<frameset>标记的属性对框架的结构进行设置。框架集的常用属性值、说明和应用举例如表 10.1 所示。

表 10.1　框架集的常用属性

参　数	说　明	举　例
COLS	在水平方向上将浏览器分割成多个窗口，取值有 3 种形式：像素、百分比（%）和相对尺寸（*）	<frameset cols="25%,100,*" > <frame></frame> </frameset>
ROWS	在垂直方向上将浏览器分割成多个窗口，取值和 COLS 类似，也是 3 种形式	<frameset rows="25%,100,*" > <frame> <frame> </frameset>
FRAMEBORDER	指定框架周围是否显示边框，取值为 1（显示边框，默认值）或 0（不显示边框）	<framset cols="25%,*" cols="*" frameborder="0"> … </frameset>
FRAMESPACING	指定框架之间的间隔，以像素为单位。默认是无间隔的	<framset cols="25%,*" cols="*" framespacing="1"> … </frameset>
BORDER	指定边框的宽度，frameborder 属性为 1 时该属性才有效	<framset cols="25%,*" cols="*" frameborder="1" border="5"> … </frameset>

3．框架属性

使用<frame>标记可以设置框架的属性，包括框架的名称，框架是否包含滚动条以及在框架中显示的网页等。<frame>标记的常用属性及其说明如表 10.2 所示。

表 10.2　框架属性

参　　数	说　　明
NAME	指定框架的名称
SRC	指定在框架中显示的网页文件（包括 HTML、PHP、JSP 等网页文件）
FRAMEBODER	指定框架周围是否显示边框，取值为 1（显示边框，为默认）或 0（不显示边框）
NORESIZE	可选属性，若指定了该属性，则不能调整框架的大小
SCROLLING	指定框架是否包含滚动条。属性可以是 yes（有）、no（没有）和 auto（自由）

10.12.3　后台首页实现过程

后台首页使用的数据表：tb_commo、tb_admin、tb_public、tb_links、tb_class

（1）定义框架页面 main.html，包含 3 个文件：top.html、left.php 和 default.php。main.html 页的代码如下：

例程43　代码位置：光盘\TM\10\ShoppingSystem\admin\main.html

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<link rel="stylesheet" href="css/style.css" />
<title>鹏逊电子商务后台管理系统</title>
</head>
<!--使用 frame 框架布局-->
<frameset rows="126,*" cols="*" frameborder="no" border="0" framespacing="0">
<!--头部框架，命名为 topFrame，载入文件为 top.html-->
  <frame src="top.php" name="topFrame" scrolling="No" noresize="noresize" id="topFrame" title="topFrame" />
  <frameset rows="*" cols="210,*" framespacing="0" frameborder="no" border="0">
<!--左侧框架，名字为 leftFrame，载入文件为 leff.php-->
<frame  src="left.php"  name="leftFrame"  frameborder="0"  scrolling="auto"  noresize="noresize"  id="leftFrame"
title="leftFrame" />
<!--   中部框架，名字为 mainFrame，载入文件为 default.php-->
    <frame src="default.php" name="mainFrame" id="mainFrame" title="mainFrame" />
  </frameset>
</frameset>
<noframes><body></body></noframes>
</html>
```

（2）left.php 页是一个树形菜单，应用 DIV+JavaScript+CSS 来实现的。首先介绍 div 标签，在 left.html 模板文件中，其关键代码如下：

例程44　代码位置：光盘\TM\10\ShoppingSystem\admin\left.html

```
<!--载入 css 样式和 JavaScript 脚本-->
<link href="css/left.css" rel="stylesheet" type="text/css" />
<script language="javascript" src="js/left.js"></script>
```



```
<!--类别管理菜单-->
<div id="type" align="center" onclick="javascript:change(one,type);">类别管理</div>
<!--子菜单-->
<div id="one" style="display: ">
<div id="addtype" align="center"><a href="addtype.php" target="mainFrame" id="menu">添加类别</a></div>
<div id="showtype" align="center"><a href="showtype.php" target="mainFrame" id="menu">查看类别</a></div>
</div>
<div id="hidediv" align="center"></div>
<!--商品管理菜单-->
<div id="commo" align="center" onclick="javascript:change(two,type);">类别管理</div>
<div id="two"style="display:none">
<!--商品管理子菜单-->
…
</div>
…
```

说明 除了加粗的 id 名称和 JavaScript 事件不同外，其他菜单的结构完全相同，此时只需修改超链接即可。

注意 除了第一个类别菜单的子菜单 display 样式为空外，其他几个子菜单的 div 样式都为 “display=none;”。

该页面在 Dreamweaver 中的效果如图 10.26 所示。

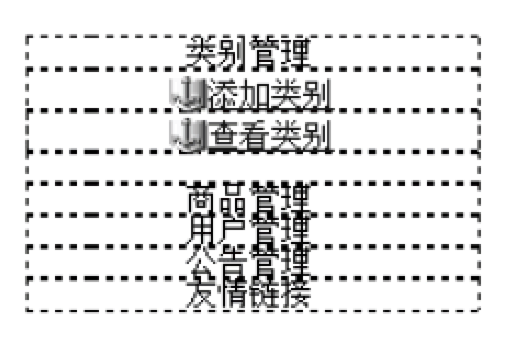


图 10.26　div 树形菜单

因为其他子菜单的样式为 display=none，所以只有“类别管理”子菜单是可见的，下面为它添加 JavaScript 事件。left.js 脚本文件的代码如下：

例程45　代码位置：光盘\TM\10\ShoppingSystem\admin\js\left.js

```
/*函数功能：改变子菜单样式，更换当前菜单的背景图片
 * nu：为子菜单 div 标签的 id 号
 * lx：为调用 change()函数所在的标签 id
*/
function change(nu,lx){
/*如果当前子菜单的样式为 none，则显示子菜单*/
	if(nu.style.display == "none"){
		nu.style.display = "";
		lx.style.background="url(images/main_openroot.gif)";
/*否则，隐藏当前菜单的子菜单*/
	}else{
```

```
            nu.style.display = "none";
            lx.style.background="url(images/main_closeroot.gif)";
}}
```

最后在 left.css 中设置 div 的长、宽等一些默认参数。至此，一个简单而又实用的树形菜单就完成了。

对于后台的大部分模块来说，其功能实现的方法和开发步骤在前台的模块设计中基本都已经介绍过。由于篇幅所限，这里不再对后台管理模块进行详细的讲解。

10.13　开发技巧与难点分析

在本系统开发和后期测试的过程中，开发人员遇到了各种各样的疑难问题。这里找出一些常见的、容易被忽略的问题加以讲解，希望能够为初学者和新手提供一些帮助，在开发程序时少走一些弯路。

10.13.1　解决 Ajax 的乱码问题

问题描述：当使用 Ajax 传递数据时，要么在数据处理页中数据不能被正确处理，要么输出返回值时显示的是一堆无法识别的乱码。

解决方法：这是因为 PHP 在传递数据时使用的编码默认为 UTF-8，这就造成了非英文字符不能正确传递的情况。解决方法如下：

在所有的 PHP 页中都输入代码“Header("Content-Type:text/html;charset=gb2312");”，这样所有的页面即可正确显示。

10.13.2　使用 JavaScript 脚本获取、输出标签内容

问题描述：获取、更改表单元素值和特定标签内容。

解决方法：使用 JavaScript 脚本获取页面内容的方式主要有两种，第一种是通过表单获取表单元素的 value 值。格式为：表单名称.元素名.value。该方式只能获取表单中的元素值，对于其他标签元素则无能为力。而第二种方式可以通过 id 名来获取页面中任意标签的内容。格式为“document.getElementById(‘id’). value;”或“document.getElementById (‘id’).innerText;”。

使用第二种方式时要注意，标签的 id 名必须存在且唯一，否则就会出现错误。为标签内容赋值时，则使用如下格式：

```
id.innerHTML ='要显示的内容';
```

10.13.3　禁用页面缓存

问题描述：使用 Ajax 技术可以防止页面刷新，但有时也会产生新的问题。例如，在会员管理页面，如果连续地冻结和解冻会员，那么超过 3 次后，该功能将失效，因为在一定时间内，如果做相同的操作，那么 xmlhttprequest 对象会执行缓存中的信息，从而造成操作失败。

解决办法：使用 header()函数将缓存关闭。将代码“header("CACHE-CONTROL:NO-CACHE");”





添加到 xmlhttprequest 对象所调用的处理页的顶部即可。

10.13.4　在新窗口中使用 session

问题描述：使用 JavaScript 文件的 open()方法打开新窗口时，原浏览器中的 session 值不会被传递到新窗口中，从而造成数据查询失败。

解决方法：将 session 值另存到隐藏域或随着 url 一起传递到新窗口。代码如下：

```
<!--在模板页中，将 session 值赋给隐藏域-->
<input id="uid" name="uid" type="hidden" value="{$smarty.session.id}">
…
/*在 JavaScript 脚本中，获取到隐藏域 value 值*/
function getInput(){
	Var uid = document.getElementById('uid').value;
/*将获取的 value 值通过 url 传给新页面*/
	open("operator.php?uid="+uid,'_blank','',false);
	…
}
```

10.13.5　判断上传文件格式

问题描述：添加商品时可以上传商品的图片，但有时可能会误传非图片格式的文件，从而导致上传图片失败。

解决方法：这里就自定义一个函数来判断上传文件的后缀，创建自定义函数 f_postfix()，函数的代码如下：

```
/*
 *判断文件后缀
 *$f_type：允许文件的后缀类型（数组）
 *$f_upfiles：上传文件名
*/
function f_postfix($f_type,$f_upfiles){
	$is_pass = false;
	$tmp_upfiles = split("\.",$f_upfiles);			//使用 split()函数分隔文件
	$tmp_num = count($tmp_upfiles);				//查找文件后缀
	if(in_array(strtolower($tmp_upfiles[$tmp_num - 1]),$f_type))	//判断后缀是否在允许列表内
		$is_pass = $tmp_upfiles[$tmp_num - 1];		//如果是，则将后缀名赋给变量
	return $is_pass;					//返回变量
}
```

10.13.6　设置服务器的时间

问题描述：如果没有对 PHP 的时区进行设置，那么使用日期、时间函数获取的将是英国伦敦本地时间（即零时区的时间）。例如，以东八区为例，如果当地使用的是北京时间，那么如果没有对 PHP 的时区进行设置，获取的时间将比当地的北京时间少 8 个小时。

解决方案：要获取本地当前的时间必须更改 PHP 语言中的时区设置。更改 PHP 语言中的时区设置有以下两种方法：

（1）在 php.ini 文件中，定位到[date]下的“;date.timezone =”选项，去掉前面的分号，并设置它的值为当地所在时区使用的时间。修改内容如图 10.27 所示。

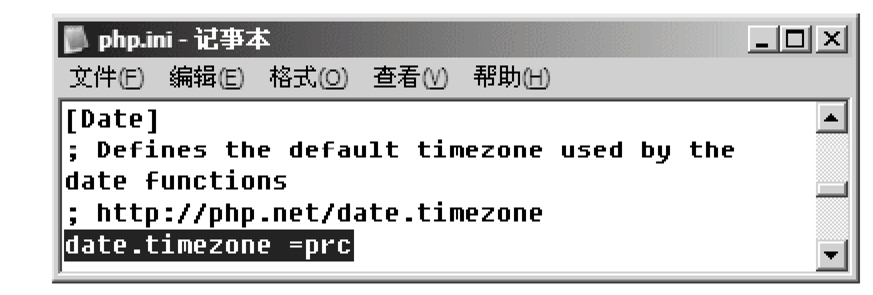


图 10.27　设置 PHP 的时区

例如，如果当地所在时区为东八区，那么就可以设置“date.timezone=”的值为 PRC、Asia/Hong_Kong、Asia/Shanghai（上海）或者 Asia/Urumqi（乌鲁木齐）等。这些都是东八区的时间。

设置完成后，保存文件，重新启动 Apache 服务器。

（2）在应用程序中，在日期、时间函数之前使用 date_default_timezone_set()函数就可以完成对时区的设置。date_default_timezone_set()函数的语法如下：

```
date_default_timezone_set(timezone);
```

参数 timezone 为 PHP 可识别的时区名称，如果时区名称 PHP 无法识别，则系统采用 UTC 时区。

例如，设置北京时间可以使用的时区包括 PRC（中华人民共和国）、Asia/Chongqing（重庆）、Asia/Shanghai（上海）或者 Asia/Urumqi（乌鲁木齐），这几个时区名称是等效的。

10.14　Smarty 类库技术专题

Smarty 是一个使用 PHP 编写的 PHP 模板引擎，是目前业界最著名、功能最强大的一种 PHP 模板引擎。它将一个应用程序分成两部分：视图和逻辑控制，也就是将 UI（用户界面）和 PHP code（PHP 代码）分离。这样，程序员在修改程序时不会影响到页面设计，而美工在重新设计或修改页面时也不会影响程序逻辑。Smarty 模板引擎的运行流程如图 10.28 所示。

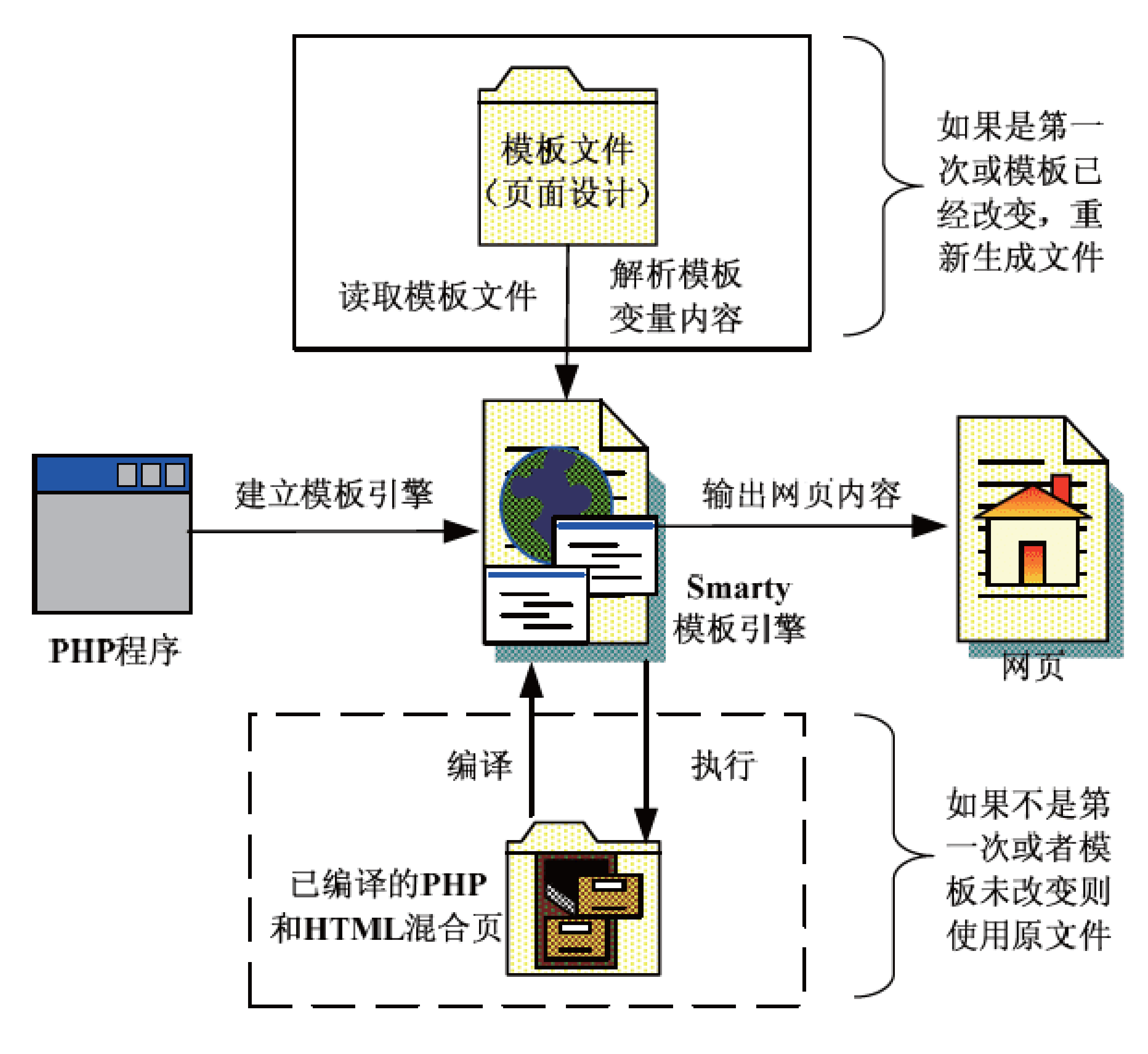


图 10.28　Smarty 模板引擎的运行流程

10.14.1　Smarty 类库下载、安装

PHP 没有内置 Smarty 模板类，需要单独下载和配置。并且 Smarty 要求服务器上的 PHP 版本最低为 4.0.6。用户可以去 http://smarty.net/download.php 下载最新的 Smarty 压缩包。本书中使用的版本是 Smarty-2.6.23，下载地址如图 10.29 所示。



图 10.29　Smarty 下载

（1）将下载的 Smarty 压缩包解压，解压后的文件夹如图 10.30 所示。其中有一个 libs 目录，这里包含 Smarty 类库的核心文件：Smarty.class.php、Smarty_Compiler.class.php、Config_File.class.php 和 debug.tpl 4 个文件，还有 internals 和 plugins 两个目录，如图 10.31 所示。

（2）Smarty 类库的载入，即将 libs 文件夹复制到服务器根目录下，并为其重新命名。一般该目录的名称为 smarty、class 等，这里将 libs 文件夹重新命名为 Smarty，并且复制到章文件夹 10\ShoppingSystem\system\smarty 下，如图 10.32 所示。



图 10.30　Smarty-2.6.23 的压缩包



图 10.31　Smarty-2.6.23 的 libs 文件

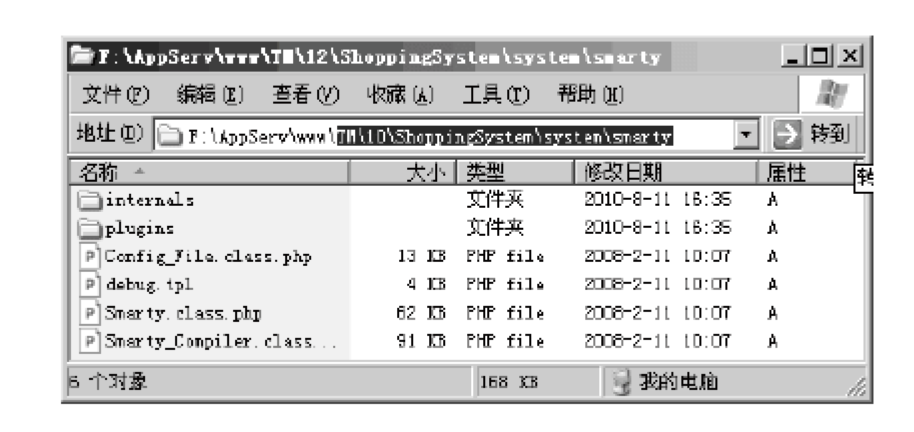


图 10.32　Smarty 类库的存储位置

10.14.2　Smarty 类库的配置

Smarty 模板的配置步骤如下：

（1）确定 Smarty 目录的位置。因为 Smarty 类库是通用的，每一个项目都可能会使用到它。所以将 Smarty 存储在根目录下，在本项目中将 Smarty 类库文件存储在 10\ShoppingSystem\system\文件夹下，命名为 Smarty。

（2）新建 templates、templates_c、configs 和 cache4 个目录。其中目录 templates 存储项目的模板

文件；目录 templates_c 存储项目的编译文件；目录 configs 存储项目的配置文件；目录 cache 存储项目的缓存文件。这 4 个目录具体放置在什么位置没有严格的规定，只要设置的路径正确，放在哪里都可以。在本项目中将创建的 4 个文件夹存储在 10\ShoppingSystem\system\文件夹下。

（3）创建配置文件。如果要应用 Smarty 模板，那么就一定要包含 Smarty 类库和相关信息。配置信息可以存储在一个单独的文件中，在需要时通过 include 语句包含即可。配置文件 config.php 的代码如下：

```
<?php
/*定义服务器的绝对路径*/
define('BASE_PATH',$_SERVER['DOCUMENT_ROOT']);
/*定义 Smarty 模板的绝对路径*/
define('SMARTY_PATH','\mr\14\Smarty\\');
/*加载 Smarty 类库文件*/
require BASE_PATH.SMARTY_PATH.'Smarty.class.php';
/*实例化一个 Smarty 对象*/
$smarty = new Smarty;
/*定义各个目录的路径*/
$smarty->template_dir = './';                                    //定义模板文件的存储位置
$smarty->compile_dir = './Smarty/templates_c/';                  //定义编译文件的存储位置
$smarty->config_dir = './Smarty/configs/';                       //定义配置文件的存储位置
$smarty->cache_dir = './Smarty/cache/';                          //定义缓存文件的存储位置
?>
```

上述配置文件的参数说明如下。

- ☑ BASE_PATH：指定服务器的绝对路径。
- ☑ SMARTY_PATH：指定 Smarty 模板的绝对路径。
- ☑ require()：加载 Smarty 类库文件 Smarty.class.php。
- ☑ $smarty：实例化 Smarty 对象。
- ☑ $smarty->template_dir：定义模板目录存储位置。
- ☑ $smarty-> compile_dir：定义编译目录存储位置。
- ☑ $smarty-> config_dir：定义配置文件存储位置。
- ☑ $smarty-> cache_dir：定义模板缓存目录。

以上是最基本的配置方法，而在本项目中使用的配置方法更加实用、灵活，在 10.14.3 节将对其进行详细讲解。

10.14.3 本项目中 Smarty 类库的配置方法

本项目中将 Smarty 配置方法封装到类中，并存储到 system.smarty.inc.php 文件中。

首先，包含 Smarty 类文件 Smarty.class.php。然后，定义 SmartyProject 类，继承 Smarty 父类。最后，定义 SmartyProject()方法，设置 Smarty 中模板文件（templates）、编译文件（templates_c）、配置文件（configs）和缓存文件（cache）的存储位置。system.smarty.inc.php 的代码如下：

```
<?php
require("../Smarty/Smarty.class.php");                           //调用 Smarty 文件
class SmartyProject extends    Smarty{                           //定义类，继承 Smarty 父类
```



```
    function SmartyProject(){                                  //定义方法，配置 Smarty 模板
        $this->template_dir = "./";                            //指定模板文件存储在根目录下
        $this->compile_dir = "./system/Smarty/templates_c/";   //指定编译文件存储位置
        $this->config_dir = "./system/Smarty/configs/";
        $this->cache_dir = "./system/Smarty/cache/";
    }
}
?>
```

既然已经将 Smarty 的配置方法存储到一个类中，那么就需要对类进行实例化，根据返回的对象名称调用 Smarty 中的方法。类的实例化操作在 system.inc.php 文件中完成，其返回对象名为$smarty，代码如下：

```
<?php
    require("system.smarty.inc.php");                          //调用类文件
    $smarty=new SmartyProject();                               //执行类的实例化操作
?>
```

通过此方法配置 Smarty 模板的好处是：无论将程序复制到哪个服务器下执行，都不需要更改服务器或者 Smarty 文件的绝对路径，程序就可以直接运行。

10.15　本 章 总 结

本章使用 Smarty、ADODB、Ajax 等目前的主流技术，实现一个电子商务平台从系统分析到最后发布的全过程。希望读者能通过这个项目实例，把前面所学到的各种技术消化吸收、融会贯通，并能够学以致用、举一反三。